Electronic Colloquium on Computational Complexity, Report No. 5 (2011)

Testing Linear Properties: Some general themes *

Madhu Sudan†

January 20, 2011

Abstract

The last two decades have seen enormous progress in the development of sublinear-time algorithms — i.e., algorithms that examine/reveal properties of "data" in less time than it would take to read all of the data. A large, and important, subclass of such properties turn out to be "linear". In particular, these developments have contributed to the rich theory of probabilistically checkable proofs (PCPs) and locally testable codes (LTCs). In this survey, we focus on some of the general technical themes at work behind the many results in this area.

Keywords: Property testing, Symmetries, Tensor Products, Error-correcting codes

1 Introduction

Property testing refers to the study of algorithms that attempt to assess properties of massive data by random sampling. The formal study views the data as some function f mapping some finite (but huge) domain D to some finite (and often small) range R. The property itself maybe specified by the set of functions $\mathcal{P}$ that satisfy the property. The central goal of property testing is to design some (probabilistic, oracle) testing algorithm that has oracle (or black-box) access to some function f, and queries f on few locations and accepts functions $f \in \mathcal{P}$ with high probability, while rejecting functions f that are *far* from $\mathcal{P}$ with high probability. Here "farness" is measured in terms of relative Hamming distance (formalized below). The first modern property tests were proposed and analyzed in the seminal works of Blum, Luby and Rubinfeld [16] and Babai, Fortnow and Lund [3]. The formal definition appeared later in the work of Rubinfeld and Sudan [43]. The first systematic study of property testing, which also extended the study beyond algebraic properties, was carried out by Goldreich, Goldwasser, and Ron [21]. Today the scope is quite extensive and the surveys by Rubinfeld [42] and Ron [41] and the articles in [20] describe some of the many developments.

In this survey, we focus on results focussing on the testability of "linear" properties. Here we restrict our attention to functions mapping to a range that is a finite field, and properties are themselves vector spaces over this field. This subcollection of properties turn out to be extremely well-motivated due to their use in the design of locally testable codes and in constructions of

*This article will also appear in the March 2011 issue of the *SIGACT News Complexity Theory Column*, edited by Lane Hemaspaandra.

†Microsoft Research New England, Cambridge, MA 02142, USA, madhu@mit.edu.

ISSN 1433-8092

probabilistically checkable proofs. We hope to highlight here some of the simple technical ideas that emerge in this line of work (and so our coverage will be spotty). Indeed, it would be more normal to consider linear properties as the same objects as locally testable codes (defined in [43], studied systematically by Goldreich and Sudan [25] and extensively since then). But we prefer to use the former term, mainly for philosophical reasons. Our interest is not only in designing codes that achieve extremal parameters (a natural tendency when speaking of codes), but to study the entire range of properties including potentially "weaker" families, mainly to get insight into when and why testability manages to emerge. Within this scope, we want to focus on some general themes, not the "best results". Nevertheless, it is good to keep "codes" in mind, since we certainly wish/need to cover them; and they offer the sharpest contrast with combinatorial properties. In testing of codes — the "members" satisfying the property are pairwise far from each other, whereas much of graph property testing considers properties where members satisying the property are clustered close to each other. Thus this area (linear property testing) ends up with really different problems, different techniques and different motivations and applications. Here we attempt to highlight three themes within: (1) Sparse properties with extremally large distance are testable; (2) Codes that are constructed by iterating a simple combinatorial operation (tensoring) are testable; (3) Properties that show sufficient symmetries are testable.

Before going on to describing these results, a personal disclaimer: As with all other surveys I've written, this one was also finished in a hurry. So errors and typos are bound to exist - but the hope is that there is more good information than bad. Furthermore, the choice of topics is obviously biased by my own research. Comments and criticisms would be most welcome.

2 Basic Definitions

We start with some of the basic definitions and properties associated with linear properties. The definitions were explored carefully in [25], but significant variety is possible and we may deviate from all known ones anyway. The basic results are mostly from the work of Ben-Sasson, Harsha and Raskhodnikova [8].

We start with the usual basic notation: We use $\mathbb{Z}$ to denote the set of integers, and $[n]$ to denote the set $\{1, \ldots, n\}$. The set $\{D \to R\}$ denotes the set of all functions from D to R. We use $\mathbb{F}_q$ to denote the finite field on q elements and $\mathbb{F}, \mathbb{K}, \mathbb{L}$ to denote arbitrary finite fields. For vector $a \in \mathbb{F}^n$, a_i with denote the ith coordinate of the vector. For vectors $a, b \in \mathbb{F}^n$, their inner product, denoted $\langle a, b \rangle$, is the element $\sum_{i=1}^{n} a_i \cdot b_i$.

For sets D and R, we use $\{D \to R\}$ to denote the set of all functions from D to R. A property $\mathcal{P}$ is a subset of $\{D \to R\}$. When the range is a finite field $\mathbb{F}$, then $\mathcal{P} \subseteq \{D \to \mathbb{F}\}$ is said to be $\mathbb{F}$-linear (or simply linear) if it is a vector space over $\mathbb{F}$. Specifically $\mathcal{P}$ is linear if for all $f, g \in \mathcal{P}$ and $\alpha, \beta \in \mathbb{F}$, the function $\alpha f + \beta g$, given by $(\alpha f + \beta g)(x) = \alpha f(x) + \beta g(x)$, is also in $\mathcal{P}$.

We use $x \leftarrow A$ to denote a random variable x chosen from distribution A. $\Pr_{x \leftarrow A}[E(x)]$ denotes the probability of event $E(x)$, and $\mathbf{E}_{x \leftarrow A}[f(x)]$ denotes the expectation of $f(x)$. We use $|A - B|$ to denote the statistical difference between distributions A and B, which is the maximum over all events E of $|\Pr_{x \leftarrow A}[E(x)] - \Pr_{x \leftarrow B}[E(x)]|$. Abusing notation somewhat, if A is just a set, then $x \leftarrow A$ will denote x drawn uniformly from A.

The normalized Hamming distance between functions $f, g : D \to R$, denoted $\delta(f, g)$, is the quantity $\Pr_{x \in D}[f(x) \neq g(x)]$. f is said to be δ-close from g if $\delta(f, g) \leq \delta$ and δ-far otherwise. We extend the notions above to the case when g is replaced by a set of functions $\mathcal{P}$ be minimizing over $g \in \mathcal{P}$. So $\delta(f, \mathcal{P}) = \min_{g \in \mathcal{P}}\{\delta(f, g)\}$ and f is *δ-close* to $\mathcal{P}$ if $\delta(f, \mathcal{P}) \leq \delta$ and *δ-far* otherwise. Finally, for a property $\mathcal{P}$, we define $\delta(\mathcal{P})$ to be the minimum over distinct functions $f, g \in \mathcal{P}$ of $\delta(f, g)$.

We are now ready to define our main definition of a locally testable property now. This definition is not the standard one, but convenient for our purposes. We clarify the difference later.

Definition 2.1 (Locally Testable Property)**.** *We say that a property $\mathcal{P} \subseteq \{D \to R\}$ is (q, ϵ, ρ)-locally testable if there is an algorithm T that makes q queries to an oracle for $f : D \to R$ and accepts $f \in \mathcal{P}$ with probability one while rejecting f that is ϵ-far from $\mathcal{P}$ with probability at least ρ. A family of properties $\{\mathcal{P}_i \subseteq \{D_i \to R_i\}\}_i$ is said to be $q(\cdot)$-locally testable if there exists some constant $\tau > 0$ and a test T that, on input i, makes $q(|D_i|)$ queries, accepts $f \in \mathcal{P}_i$ with probability 1 and rejects every other f with probability $\tau \cdot \delta(f, \mathcal{P}_i)$.*

We now comment on the distinctions with standard definitions. (A reader unfamiliar with related definitions can safely skip this and the next two paragraphs.) First, the definition above only allows for "one-sided" error, but in the case of linear properties one can get this feature (and others, see below) without loss of generality.

More significantly, this definition allows only "proximity oblivious tests" in the sense of Goldreich and Ron [23]. To explain this, the usual definition of testing allows a "proximity parameter" ϵ as input to the tester and allows the query complexity to grow as $\epsilon \to 0$, while requiring the tester to reject only functions that are ϵ-far with some fixed probability. In our case, the tester is not promised any lower bounds on the proximity of the function being tested. It must simply pick its query points oblivious of this parameter and decide whether to accept or not. It is natural in this setting that functions very close to having the property may evade detection, and so our requirement that functions are rejected with probability proportional to their distance from the property. In the setting of combinatorial and graph properties, proximity oblivious tests are not the most natural and required separate study. In the case of linear properties, we are not aware of any proximity-sensitive tests and indeed many basic concepts (such as constraints and characterizations, see below) are naturally related to proximity-oblivious tests, and so we restrict ourselves to this setting.

Finally, the test above is also a "strong" test in that it is required to reject every function not in $\mathcal{P}$ with non-zero probability. Again this is somewhat restrictive, but all known tests satisfy this condition and since we are mostly interested in "positive results" it is good to get this strong property.

We now move to some basic "combinatorial" way of looking at local tests for linear properties.

Definition 2.2 (Constraints, Characterizations)**.** *For functions mapping D to $\mathbb{F}$ a k-*local constraint *C is a collection of points $\alpha_1, \ldots, \alpha_k \in D$ and a vector space $V \subsetneq \mathbb{F}^k$. A function f satisfies C if $(f(\alpha_1), \ldots, f(\alpha_k)) \in V$. A property $\mathcal{P}$ satisfies C if every function in $\mathcal{P}$ satisfies C. We say that C is a* basic constraint *if V is given by a single linear equation $\sum_i \lambda_i f(\alpha_i) = 0$ with λ_i's being non-zero. A collection of constraints $C_1, \ldots, C_m$ give a k-*local characterization *of $\mathcal{P}$ if every C_j is k-local and $f : D \to R$ is in $\mathcal{P}$ if and only if it satisfies C_j for every j.*

The following proposition from [8] shows the intimate relationship between local tests and constraints and characterizations. To describe their proposition, we first define the notion of a *canonical q-tester* T for a property $\mathcal{P} \subseteq \{D \to \mathbb{F}\}$. We say T is a *canonical q-tester* if it is given by a collection of q-local basic constraints $C_1, \ldots, C_m$ and distribution A on $[m]$. Given oracle acces to f, the tester T simply picks $j \leftarrow A$ and accepts f if and only if it satisfies C_j.

Proposition 2.3 ([8]). *Let $\mathcal{P} \subseteq \{D \to \mathbb{F}\}$ be a linear property with a q-query testing algorithm accepting $f \in \mathcal{P}$ with probability at least c while accepting f that is ϵ-far from $\mathcal{P}$ with probability at most $c - \rho$. Then $\mathcal{P}$ has a* canonical tester *T which accepts $f \in \mathcal{P}$ with probability one, while accepting f that is ϵ-far with probability at most $1 - (1 - 1/|\mathbb{F}|) \cdot \rho$.*

Note that canonical testers are special in that they are non-adaptive (they decide on all their queries before examining the responses to other queries) and makes one-sided error only. Furthermore, the final test is extremely simple: It simply checks that the responses satisfy a single homogenous linear equation, and all for a small price in the error and no loss in query complexity!

Finally, if the original tester satisfies the strong condition that every function not in $\mathcal{P}$ is rejected with positive probability, then the resulting collection of constraints implied by the canonical tester form a q-local characterization. We remark that a property with a local characterization is a well-studied concept under the label of "low-density parity check (LDPC) codes".

An early hope that every LDPC code may be testable was eventually refuted by [8] who showed that even "random" LDPC codes are not locally testable. Further explanation for why this happens was given recently by Ben-Sasson et al. [7] who show that locally testable codes need to have "redundancy" among its local constraints. Specifically, if one examines the constraints $C_1, \ldots, C_m$ in the support of the distribution A in Proposition 2.3, it must be the case that a constant fraction of such constraints must be implied by the remaining ones for the property to be $O(1)$-locally testable. The notion of redundancy turns out to give useful insight into local testability, and in each of the cases in the following sections we will attempt to explain where this comes from (first).

With these preliminaries in places, we are ready to start talking about some themes in testing.

3 Sparse Properties: "Structure Everywhere"

The first general class of locally testable linear properties are what we call "sparse" ones. Here we consider properties on a domain of size N, with only $N^{O(1)}$ functions satisfying the property. Furthermore, if functions in the property have pairwise distance at least $1/2 - N^{-\Omega(1)}$ then it turns out any such property is locally testable. One motivation for this was to generalize the linearity test in [16] which ends up showing that the "sparsest possible", "highest-distance possible" linear property is testable. We will describe this interpretation of their result below, after giving the main definitions and theorem of the section.

We start with a basic observation about linear properties:

Proposition 3.1. *For every linear property $\mathcal{P} \subseteq \{D \to \mathbb{F}_2\}$, we can view the domain D as a subset of $\mathbb{F}_2^n$ (for some appropriate n, which we refer to as the dimension of $\mathcal{P}$) and for every function $f : D \to \mathbb{F}_2$, $f \in \mathcal{P}$ if and only if there exists an $a \in \mathbb{F}_2^n$ such that $f(x) = \langle a, x \rangle$.*

Thus every linear property $\mathcal{P}$ with range $\mathbb{F}_2$ is described by its dimension n and the domain $D \subseteq \mathbb{F}_2^n$. We use $\mathcal{P}_D$ to denote the property described by the domain $D \subseteq \mathbb{F}_2^n$.

We say that a property $\mathcal{P} \subseteq \{D \to \mathbb{F}\}$ is *c-sparse* if $|\mathcal{P}| \leq |D|^c$. Note that if $\mathcal{P}$ has dimension n, then $|\mathcal{P}| = 2^n$ and for a c-sparse property, this implies $|D| \geq 2^{n/c}$. The main result of this section is stated below.

Theorem 3.2 (Kaufman and Sudan [34])**.** *For every $\gamma > 0$ there exists a $q < \infty$ such that for every n and every linear property $\mathcal{P} \subseteq \{D \to \mathbb{F}_2\}$ of dimension n that is c-sparse and satisfies $\delta(\mathcal{P}) \geq \frac{1}{2} - 2^{-\gamma \cdot n}$, $\mathcal{P}$ is q-locally testable.*

Note that the condition that $\mathcal{P}$ is $\delta(\mathcal{P}) \geq \frac{1}{2} - 2^{-\gamma \cdot n}$ actually implies $c \leq \frac{1}{\gamma}$. The proof in [34] of the above theorem relied principally on techniques introduced earlier by Kaufman and Litsyn [30] who also gave a (quantitatively and qualitatively) weaker version the theorem. In this section however, we will give a much simpler proof of the qualitatively weaker version of the theorem above, from the work of Kopparty and Saraf [37].

The result from [37] only works for "small-biased" properties (as opposed to high-distance ones). We define this concept next. A property is said to be *ϵ-biased* if for every pair of distinct functions $f, g \in \mathcal{P}$, we have $|\delta(f, g) - \frac{1}{2}| \leq \epsilon/2$. [37] proves this result for the case of properties $\mathcal{P}$ whose bias is at most $2^{-\gamma n}$. We focus on the weaker result since it seems almost as interesting, and the proofs are nicer.

We will sketch the proof of Theorem 3.2 later, but first we describe the "ultimate" special case of this theorem, namely the Hadamard property, which is important as motivation and also as an ingredient in the proof.

3.1 Hadamard Property

Had is the n-dimensional property $\mathcal{P}_D$ for the special case of D being the entire set $\mathbb{F}_2^n$. In other words, the Hadamard property consists of all the linear functions mapping $\mathbb{F}_2^n$ to $\mathbb{F}_2$. It is the ultimate property in various senses, as described below.

Proposition 3.3. *1. Every linear property $\mathcal{P}$ is a "subcode" of the Hadamard property.*

2. The Hadamard property is 1*-sparse.*

3. The Hadamard property is 0*-biased.*

Part (1) of the Proposition is just a restatement of the definition, Part (2) follows immediately from the definition of sparsity, and Part (3) is a standard coding-theory fact (whose proof we leave to the reader as an exercise).

[16] shows that the following 3-query test is a good one for the Hadamard property: "Pick $x, y \in_U \mathbb{F}_2^n$ and accept f if and only if $f(x)+f(y) = f(x+y)$." We refer to this test as the BLR-test. Subsequent work of Bellare et al. [5] gives a slightly cleaner statement which we describe below.

Theorem 3.4 ([16, 5])**.** *The BLR-test accepts $f \in$ Had with probability one, while rejecting all f with probability at least $\delta(f, \text{Had})$.*

Given Proposition 3.3, it becomes clear that the theorem above yields Theorem 3.2 for the "ultimate" special case. The remarkable aspect of [37] is that it manages to reduce the general case to this special case, as we describe next.

3.2 Reducing sparse testing to Hadamard testing

The key idea behind the test and analysis in [37] is to consider a t-fold "direct sum" (probabilistic) function $f^{(\oplus t)} : \mathbb{F}_2^n \to \mathbb{F}_2$ which is defined from f and to show that if $f \in \mathcal{P}$ then $f^{\oplus t} \in \mathrm{Had}$ whereas $\delta(f^{(\oplus t)}, \mathrm{Had})$ is bounded away from 0 if $\delta(f, \mathcal{P})$ is bounded away from 0. In order to make their argument precise, we need to define the direct sum function and extend the notion of distances to probabilistic functions.

We start by recalling two basic probability facts that are commonly used in TCS research (so we won't give exact references). First we note that if $x_1, \ldots, x_t$ are independent random variables taking on values in $\mathbb{F}_2$ with $\Pr[x_i = 0]$ being $(1+\alpha)/2$, then the probability that $\sum_i x_i = 0$ (sum being over $\mathbb{F}_2$) is exactly $(1+\alpha^t)/2$. Next we recall the "Vazirani XOR lemma" which says that for a random variable x taking values in $\mathbb{F}_2^n$ with some distribution A, if it holds that for every $a \in \mathbb{F}_2^n$, $|\Pr_{x \leftarrow A}[\langle a, x\rangle = 0] - 1/2| \leq \epsilon$, then $|A - U| \leq \epsilon 2^n$ where U denotes the uniform distribution on $\mathbb{F}_2^n$.

We now return to our task. Recall we are considering a function $f : D \to \mathbb{F}_2$. Let $D^{(*t)}$ denote the distribution on $\mathbb{F}_2^n$ sampled by picking $(x_1, \ldots, x_t) \leftarrow D^t$ and outputting $x_1 + \cdots + x_t$. For x in the support of $D^{(*t)}$ let D_x^t denote the distribution on $(\mathbb{F}_2^n)^t$ given by picking $(x_1, \ldots, x_t)$ according to D^t, conditioned on $\sum_{i=1}^t x_i = x$.

For $f : D \to \mathbb{F}_2$, let $f^{(\oplus t)} : \mathbb{F}_2^n \to \mathbb{F}_2$ be the probabilistic function given as follows: For $x \in \mathbb{F}_2^n$, if x is in the support of $D^{(*t)}$, then select $(x_1, \ldots, x_t) \leftarrow D_x^t$ and output $f(x_1) + \cdots + f(x_t)$. [If x is not in the support of $D^{(*t)}$, we may set $f^{(\oplus t)}(x) = 0$, though this case will not arise in the following.]

We are now in a position to describe the test in [37].

1. Pick t to be an odd integer that is sufficiently large (depending on γ).
2. Pick $x, y \leftarrow \mathbb{F}_2^n$.
3. Sample (independently) $a = f^{(\oplus t)}(x)$, $b = f^{(\oplus t)}(y)$ and $c = f^{(\oplus t)}(x+y)$ and accept if $a+b = c$.

To analyze the test, one more definition: We extend the definition of distance between functions also to probabilistic functions. So for probabilistic functions $f, g : S \to \mathbb{F}_2$, let $\delta(f, g) = \mathbf{E}_{x \in_U S}[|\Pr[f(x) = 1] - \Pr[g(x) = 1]|]$. (Note that this does extend our usual definition.)

The key claims that allow an analysis of the test above are the following: (1) As $t \to \infty$, the distribution $D^{(*t)}$ converges to the uniform distribution on $\mathbb{F}_2^n$ (proved using the Vazirani XOR lemma and the small-bias of $\mathcal{P}$), (2) If $f \in \mathcal{P}$ then $f^{(\oplus t)}$ converges to a member of the Hadamard property (verified easily) and (3) If $\delta(f, \mathcal{P}) = \frac{1-\alpha}{2}$ then $\delta(f^{(\oplus t)}, \mathrm{Had}) \approx \frac{1-\alpha^t}{2} \pm |D^{(*t)} - U|$ where U is the uniform distribution on $\mathbb{F}_2^n$ (this fact is proved using the other probability fact mentioned above). Once these three facts are shown, the analysis of the test is immediate!

Before concluding this section, let us just note where the "redundancy" of local constraints (which we claimed was necessary for testability) occurs here. The claim that $D^{(*t)}$ converges to the uniform

distribution implies that for every $x \in \mathbb{F}_2^n$ there are many t-tuples in D^t, that sum to x. Any pair of these, say $\alpha_1, \ldots, \alpha_t$ and $\beta_1, \ldots, \beta_t$, gives a constraint (of locality $2t$) since if $f(\cdot) = \langle a, \cdot \rangle$, then $f^{(\oplus t)}(x) = \sum_{i=1}^{t} f(\alpha_i) = \sum_{j=1}^{t} f(\beta_j)$. So the proofs are effectively showing that redundancy follows from some (simple) counting. This of course relies heavily on the sparsity; and so in following sections we seek other sources of redundancy.

4 Tensor Products: "Testability by Design"

While the previous section focused on the testability of "random" or "natural" codes/properties, of appropriate parameters, most of the "efficient" codes tend to be designed by a series of intricate operations. In this section we focus on one of the simplest operation that manages to build properties with a redundant collection of local constraints, and (in some cases) use this redundancy to design/analyze local tests for these properties. Our initial codes will only be "mildly" locally testable (say with $\sqrt{|D|}$ sized tests), and even the final ones will be $\omega(1)$-locally testable. However we note that the techniques can/do play a role in some of the best known constructions of $O(1)$-locally testable codes due to Meir [38]. With these preliminaries in place, lets get down to business.

4.1 Tensor Products of Properties

Definition 4.1. *Given two properties $\mathcal{P}_1 \subseteq \{D_1 \to \mathbb{F}\}$ and $\mathcal{P}_2 \subseteq \{D_2 \to \mathbb{F}\}$, their tensor product, $\mathcal{P}_1 \otimes \mathcal{P}_2 \subseteq \{D_1 \times D_2 \to \mathbb{F}\}$, is the property*

$$\mathcal{P}_1 \otimes \mathcal{P}_2 = \{f : D_1 \times D_2 \to \mathbb{F} | \forall a \in D_1, f(a, \cdot) \in \mathcal{P}_2 \text{ and } \forall b \in D_2, f(\cdot, b) \in \mathcal{P}_1\}.$$

A priori it may not be clear that the tensor product is a non-empty property, and indeed for non-linear properties it may not be. However for linear properties it is a nice notion, and below we mention some notions that help explain why.

For a property $\mathcal{P} \subseteq \{D \to \mathbb{F}\}$, a set $S \subseteq D$ is said to be an *interpolating* set if for every function $f : S \to \mathbb{F}$ there exists a unique *extension* $\hat{f} \in \mathcal{P}$, i.e., $\hat{f} : D \to \mathbb{F}$ and satisfies $\hat{f}(a) = f(a)$ for every $a \in S$.

The following three propositions follow from basic linear algebra.

Proposition 4.2. *Every linear property $\mathcal{P}$ has an interpolating set.*

A useful fact relating interpolating sets to distances of codes is the following.

Proposition 4.3. *Every set $T \subseteq D$ satisfying $|T| > (1 - \delta(\mathcal{P})) \cdot |D|$ contains an interpolating set for $\mathcal{P}$.*

The next proposition, while also elementary, is a very powerful feature of tensor products.

Proposition 4.4. *For $i \in \{1, 2\}$ if S_i is an interpolating set for $\mathcal{P}_i$, then $S_1 \times S_2$ is an interpolating set for $\mathcal{P}_1 \otimes \mathcal{P}_2$.*

The importance of the proposition above to this section is that it implies a certain redundancy among the constraints specifying tensor products. Let us elaborate on this. Suppose we are given a function $f : S_1 \times S_2 \to \mathbb{F}$. To "extend" it to a function $\hat{f} : D_1 \to D_2 \to \mathbb{F}$ in $\mathcal{P}_1 \times \mathcal{P}_2$, we could do so in two steps. First we could extend f to a function $\tilde{f} : D_1 \times S_2 \to \mathbb{F}$, by fixing $b \in S_2$ and letting $\tilde{f}(\cdot, b)$ be the extension from $\mathcal{P}_1$ of $f(\cdot, b)$ as guaranteed to exist by the interpolating property of S_1. Next we could let $\hat{f} : D_1 \to D_2 \to \mathbb{F}$ be the function obtained by extending, for every $a \in D_1$, $\tilde{f}(a, \cdot)$ to the function $\hat{f}(a, \cdot) \in \mathcal{P}_2$ (now using the interpolating property of S_2). The uniqueness of the interpolating steps makes it clear that the resulting function is the only one that could satisfy the conditions $\tilde{f}(a, \cdot) \in \mathcal{P}_2$ for every $a \in D_1$ and $\tilde{f}(\cdot, b) \in \mathcal{P}_1$ for every $b \in S_2$, which are necessary to ensure $\tilde{f} \in \mathcal{P}_1 \otimes \mathcal{P}_2$. But the definition includes more constraints, since we also need to satisfy the conditions $\tilde{f}(\cdot, b) \in \mathcal{P}_1$ for $b \notin S_2$. The remarkable (though simple) fact about linear properties is that these conditions are actually redundant and will be automatically satisfied by the construction above. In the quest for "operations" that support testability, such a natural emergence of redundancy is very encouraging, and motivate the question of the testability of tensor products of codes.

4.2 Robust Local Testability

A priori, just the redundancy mentioned above need not lead to local testability. But we in fact seek even more, something called "robust local testability". To motivate this notion, let us reveal our agenda. Our goal really is to start with some small code (i.e., property on small domain) with moderate local testability, and then take its tensor with itself many times to get a really long code with local testability being roughly the same as that of the starting code! And furthermore we hope to analyze the testability by induction. So suppose we've managed to show that $\mathcal{P}^{\otimes k}$ is locally testable and we would now like to show that $\mathcal{P}^{\otimes(2k)} = \mathcal{P}^{\otimes k} \otimes \mathcal{P}^{\otimes k}$ is also locally testable. And the test we hope to analyze is the following: Given f, pick $a, b \leftarrow D^k$ uniformly and verify $f(a, \cdot)$ and $f(\cdot, b)$ are in $\mathcal{P}^{\otimes k}$. We'd like a statement about tensor products to show that such a test implies f is contained in the tensor product. Unfortunately, the local tests can only show that $f(a, \cdot)$ and $f(\cdot, b)$ are *close* to being members of $\mathcal{P}^{\otimes k}$ and not that they are actually members of this set. So what we'd really like is a statement of the form "If $f(a, \cdot)$ and $f(\cdot, b)$ are usually close to $\mathcal{P}_1$ and $\mathcal{P}_2$, then f is close to $\mathcal{P}_1 \times \mathcal{P}_2$." Such a property is what we formally define as robust testability below.

Definition 4.5. *The tensor of properties $\mathcal{P}_1$ and $\mathcal{P}_2$ is said to be* α-robust locally testable *(or simply α-robust) if the following holds for every function $f : D_1 \times D_2 \to \mathbb{F}$:*

$$\delta(f, \mathcal{P}_1 \otimes \mathcal{P}_2) \leq \alpha \cdot (\mathbf{E}_{b \leftarrow D_2}[\delta(f(\cdot, b), \mathcal{P}_1)] + \mathbf{E}_{a \leftarrow D_1}[\delta(f(a, \cdot), \mathcal{P}_2)]) .$$

The initial hope expressed in Ben-Sasson and Sudan [10] was that perhaps the tensor of any pair of high-distance properties may be α-robust (where α depends only of the relative distance, but not the size of the domain). But this was refuted with a very creative counterexample by P. Valiant [45]. In view of this counterexample, one has to consider more restricted classes of properties, or other tests. We describe some of the positive results next.

4.3 Robust testability results

The first robustness result for a "tensor" was a very specific one, where the properties $\mathcal{P}_1$ and $\mathcal{P}_2$ were the properties of being low-degree univariate polynomials. Specifically, let $\mathcal{RS}(d,\mathbb{F}) \subseteq \{\mathbb{F} \to \mathbb{F}\}$ consist of all polynomial functions in $\mathbb{F}[x]$ of degree at most d. The tensor of $\mathcal{RS}(d,\mathbb{F})$ with itself consists of all bivariate functions $f(x,y)$ that are polynomials of degree at most d in x and at most d in y. Robustness of the tensor would imply that if a bivariate function usually is in good agreement with a univariate polynomial (for random settings of its first or second variable), then the funcion must be close to a bivariate polynomial. This innocuous statement turns out to be quite non-trivial to analyze. Early results (e.g., from [43]) could only yield robustness depending on d. Subsequent results due to Arora and Safra [2] and Polishchuk and Spielman [39] however improved this significantly leading to the following theorem.

Theorem 4.6 ([39])**.** *There exist constants c, α such that for every field $\mathbb{F}$ and $d \leq |\mathbb{F}|/c$, the tensor $\mathcal{RS}(d,\mathbb{F}) \otimes \mathcal{RS}(d,\mathbb{F})$ is α-robust.*

Of course, given the specific algebraic nature of the results, they were not stated in terms of tensor products. The explicit study of robustness of tensor products (motivated partially by an attempt, thus far unsuccessful, to generalize the above theorem) started in [10]. In their work, however they considered a slightly different class of properties and tests. Rather than testing the tensor of two properties with the projection to one of the two dimensions, they considered the tensor of three properties and tested them by projecting onto various two dimensional surfaces. We define the resulting robustness concept next.

Definition 4.7. *The triple-wise tensor of a property $\mathcal{P}$ with itself said to be* α-pairwise robust *if the following holds for every function $f : D \times D \times D \to \mathbb{F}$:*

$$\delta(f, \mathcal{P} \otimes \mathcal{P} \otimes \mathcal{P}) \leq \alpha \cdot \mathbf{E}_{a \leftarrow D} \left[\delta(f(\cdot,\cdot,a),\mathcal{P}) + \delta(f(\cdot,a,\cdot),\mathcal{P}) + \delta(f(a,\cdot,\cdot),\mathcal{P})\right].$$

[10] shows that if $\delta(\mathcal{P})$ is large enough, then $\mathcal{P}^{\otimes 3}$ has positive robustness.

Theorem 4.8. *There exists $\delta < 1$ and $\alpha < \infty$ such that for every property $\mathcal{P}$ with $\delta(\mathcal{P}) \geq \delta$, $\mathcal{P}^{\otimes 3}$ is at least α-pairwise robust.*

We note that properties with arbitrarily large $\delta < 1$ exist provided the range $\mathbb{F}$ is sufficiently large. In particular, random linear codes, as well as algebraic codes like Reed-Solomon or algebraic-geometric codes over large alphabets can satisfy the requirement.

We won't prove the theorem above but the main idea, which goes back to a work of Raz and Safra [40] is as follows: Given a function $f \notin \mathcal{P}^{\otimes 3}$, we focus on the combinatorial properties of a graph derived from f. The graph has $3 \cdot |D|$ vertices, one corresponding to each "plane" (i.e., the set of points of the form $\{(a,\cdot,\cdot)\}$, or $\{(\cdot,b,\cdot)\}$ or $\{(\cdot,\cdot,c)\}$). To each vertex we associate the nearest member of $\mathcal{P} \otimes \mathcal{P}$ (so for the plane $\{(a,\cdot,\cdot)\}$ we associate a function $g_a \in \mathcal{P} \otimes \mathcal{P}$ which minimizes the distance to $f(a,\cdot,\cdot)$). We then throw away vertices where the distance between $f(a,\cdot,\cdot)$ and g_a is noticeably high (based on a carefully chosen threshold). We now add edges to this graph to capture "inconsistencies". Specifically we add an edge between the a-plane ($\{(a,\cdot,\cdot)\}$) and b-plane ($\{(\cdot,b,\cdot)\}$), if the functions g_a and g_b are in disagreement on some point of the form (a,b,x). The

"interpolating property" of tensor products tell us that if this graph has a large independent set consisting of $(1-\delta)$ fraction of planes in each of the three coordinate axes, then there is some function $g \in \mathcal{P}^{\otimes 3}$ which is consistent with the g_a's on this independent set, and is hence close to f. So the principal question reduces to showing that this graph has a large independent set. To this end, it is easy to show that the graph is relatively sparse (has only τ-fraction of all edges, for arbitrarily small, but constant τ). But this is not sufficient to imply a large independent set (in random graphs of this density the size of the independent set may only be constant depending on $1/\tau$). Here is where the structure of the tensor product code comes into play. Suppose there is an edge between the a-plane and the b-plane. Then the functions g_a and g_b, when restricted to the "line" $\{(a, b, \cdot)\}$ are different members of $\mathcal{P}$ and so disagree on most points. It follows that for most choices of c, either the a-plane is adjacent to the c-plane or the b-plane is adjacent to the c-plane. We conclude that at least one of two endpoints must have (linearly) high degree. So, if we throw away from the graph all vertices with linearly high degree, then we are left with an independent set (and as argued above, this suffices)!

Moving on, we now describe a somewhat different approach to getting broad robustness results for tensor products, as studied by Dinur, Sudan and Wigderson [19]. They considered the original tensor product of *two* codes, but restricted (one of) the codes to have "nice" locality properties. E.g., one of the codes being tensored may itself be locally testable, or one of the codes may be a "low density parity check" code. In such cases they show that the tensor does show robustness. These results were subsequently strengthened significantly in the works of Ben-Sasson and Viderman [14, 13]. Below we state a sample result in this area. (We don't state the most general results, since stating them requires more definitions. The reader is pointed to the works for complete results.)

Theorem 4.9 ([13, Theorem 10])**.** *For every $\epsilon, \delta, \rho > 0$, and $q < \infty$ there exists an $\alpha < \infty$ such that if $\mathcal{P}_1, \mathcal{P}_2$ satisfy $\delta(\mathcal{P}_1), \delta(\mathcal{P}_2) \geq \delta$ and $\mathcal{P}_1$ is (q, ϵ, ρ)-LTC, then $\mathcal{P}_1 \otimes \mathcal{P}_2$ is α-robust.*

Again we won't prove the result, but let us highlight the main idea behind the proof. Consider a function $f : D_1 \times D_2 \to \mathbb{F}$, for which the "testing error", i.e., the quantity $\mathbf{E}_{b \leftarrow D_2}[\delta(f(\cdot, b), \mathcal{P}_1)] + \mathbf{E}_{a \leftarrow D_1}[\delta(f(a, \cdot), \mathcal{P}_2)]$ is small. We'd like to show f is close to a member of $\mathcal{P}_1 \times \mathcal{P}_2$. Let f_1, f_2 be the "row" and "column"-wise decoding of f. I.e., $f_1(\cdot, b) \in \mathcal{P}_1$ for every b and among all such functions, it is one that minimizes $\delta(f, f_1)$. Similarly f_2 is the nearest function to f satisfying $f_2(a, \cdot) \in \mathcal{P}_2$ for every a. Now consider the "error" function $e = f_1 - f_2$. It can be easily seen that $\delta(e, 0) \leq \mathbf{E}_{b \leftarrow D_2}[\delta(f(\cdot, b), \mathcal{P}_1)] + \mathbf{E}_{a \leftarrow D_1}[\delta(f(a, \cdot), \mathcal{P}_2)]$ and so e is rarely non-zero. The key step in the analysis is to show that there are large subsets (large enough to contain interpolating sets) $S_1 \subseteq D_1$ and $S_2 \subseteq D_2$ such that e is identically zero on $S_1 \times S_2$. One hope would be that once we throw away from D_1 and D_2 rows and columns where e is non-zero on a noticeable fraction of points, e becomes all zero on the remaining points. When $\mathcal{P}_1$ is an LTC, then something nice happens. Suppose there is a "basic local test" of $\mathcal{P}_1$ that examines a given function at locations $i_1, \ldots, i_q \in D_1$ and verifies that the sum of the values of the function at these locations is zero. Suppose further that $e(i_1, \cdot), \ldots, e(i_q, \cdot)$ are rarely non-zero. In particular suppose $\delta(e(i_j, \cdot), 0) < \delta/q$. Then it turns out (by a simple coding-theoretic argument) that $\sum_{j=1}^{q} e(i_j, \cdot) = 0$. (So the sum of the error values on these rows is not just rarely non-zero, it is identically zero.) In other words, every "column" of e satisfies this constraint. A simple counting argument reveals that the e function satisfies such constraints for many of the local tests of $\mathcal{P}_1$ and the local testability of $\mathcal{P}_1$ now implies that once we erase a small set of rows of e, it would look like a member of $\mathcal{P}_1 \times \mathcal{P}_2$ on every column. In turn this implies on the non-erased rows, most columns are zero (since e is usually zero).

4.4 Usage of tensor products

Based on just simple repetition of the tensor operation one can design codes of block length n that have $\Omega(1)$ distance and dimension $n^{\Omega(1)}$ that are testable with $\operatorname{poly}\log n$ queries [10, Theorem 2.7]. Indeed the above work (see also [13]) shows that one can start with any high-distance code and tensor it enough times to get such locally testable codes.

However, this is weak compared to the best known locally testable codes in the literature. The best known performance yields codes of distance $\Omega(1)$ with dimension $n/\operatorname{poly}\log n$ that are testable with $O(1)$ queries, as given by Dinur [17] building on codes with same distance and dimension with $\operatorname{poly}\log n$ queries given by Ben-Sasson and Sudan [11]. It turns out even this optimal performance can be matched by codes constructed by mostly combinatorial steps, with the most "algebraic" operation being that of taking tensors, as shown by Meir [38]. Indeed theorems such as those listed above do play some role in Meir's construction, though there are many other careful operations that are used and analyzed to get the final result.

5 Invariance: "Testability in Nature"

The previous sections focussed on two possible themes that could lead to testability: Either we pick parameters to be so extremal that testing is inevitable, or we design codes so carefully that testing can be imposed. In this section we consider a third option: that the property offers enough structure/symmetry as to allow testing. To this end let us explain what we mean by symmetries and then focus on a specific type of symmetry which seem appropriate to study (in that they are commonly seen) and the current understanding we have of properties with this class of symmetries.

5.1 Invariance in Property Testing

We say that $\mathcal{P} \subseteq \{D \to \mathbb{F}\}$ is *invariant* under a function $\pi : D \to D$ if for every $f \in \mathcal{P}$ it is the case that the function $f \circ \pi$, defined as $f \circ \pi(x) = f(\pi(x))$, is also in $\mathcal{P}$. We say that $\mathcal{P}$ is invariant under a set $G \subseteq \{D \to D\}$ if for every $\pi \in G$, $\mathcal{P}$ is invaraint under π. The set of all functions π under which $\mathcal{P}$ is invariant is termed the invariance class of $\mathcal{P}$. (The invariance class is a semi-group under composition.) The set of all *permutations* (bijections) π under which $\mathcal{P}$ is invariant is the *automorphism group* of $\mathcal{P}$.

The notion of examining testability of properties with explicit attention on their invariance is a slowly emerging theme. An early result of Babai, Shpilka and Stefankovic [4] gave lower bounds on rates of locally testable codes for cyclic codes is perhaps the first to explicitly relate testability to invariances, albeit to give negative results. The work by Goldreich and Sheffet [24] also uses symmetries to give lower bounds on query complexity. Alon et al. [1] were possibly the first to suggest this might lead to positive results. The work by Kaufman and Sudan [33] seems to be the first to to explicitly focus on invariances to derive positive results. The class of invariances they studied were "linear-invariances" and "affine-invariances". We will describe these shortly, but before doing so, we will first point out the broader relevance of invariances in property testing.

While invariances were not explicitly highlighted in works (other than those listed above), a lot of the work in property testing does assume and exploit invariances. Indeed all of "graph property testing"

(an extensively studied class of properties) is characterized by its "invariances" — a property is a graph propery if and only if it is invariant under all vertex renamings. Similarly "properties of Boolean functions" are those that are invariant under renamings of variables. "Statistical" properties in turn are those that are invariant not only under (all) permutations of the domain, but also under permutations of the range. Each of these classes is an extensively studied class of properties. We won't list the results here, but the reader is pointed to [44, Section 2] for a more detailed description and pointers to these lines of work. We merely wish to assert that there is pedagogical benefit to viewing all of these results as being characterized by the nature of invariance assumed and consequences derived.

5.2 Affine and Linear Invariance

The invariances of greatest interest to "algebraic property testing" are those induced by linear or affine transformations of the domain. To motivate this, notice that one of the basic properties of interest in property testing is that of a function being a low-degree multivariate polynomial. This property is invariant under any affine transformation of the domain: E.g., the degree of the polynomial $p(3x+y, y-x)$ is at most the degree of $p(x, y)$. Furthermore, examination of the known tests of this property makes it clear that this aspect is central to the tests. This motivates some natural questions: Does this symmetry (invariance under affine transformations) suffice to explain the tests? What other properties are invariant under such transformations? Can any of the others lead to codes with better performance than multivariate polynomials? Such questions motivate a systematic study of affine-invariance and this has been carried out in a sequence of works. We report on the main results below.

First start with the formal setup. From this point onwards, throughout this section we will consider functions from an n-dimensional vector space over a field $\mathbb{K}$ of size 2^s to a subfield $\mathbb{F}$. For simplicity we will just set $\mathbb{F} = \mathbb{F}_2$ (though much of the study can be (and has been) extended to fields of arbitrary finite characteristic). Looking forward, we also note that pretty soon we would have managed to restrict our attention to the case $n = 1$ which turns out to be the most general one.

We will consider linear properties $\mathcal{P}$ of functions mapping $\mathbb{K}^n \to \mathbb{F}$ that are "affine-invariant" in the sense below.

Definition 5.1 (Affine-invariant Properties)**.** *A function $A : \mathbb{K}^n \to \mathbb{K}^n$ is said to be affine if there exists a matrix $M \in \mathbb{K}^{n\times n}$ and a vector $b \in \mathbb{K}^n$ such that $A(x) = Mx + b$ for $x \in \mathbb{K}^n$. A property $\mathcal{P} \subseteq \{\mathbb{K}^n \to \mathbb{F}\}$ is said to be affine-invariant (over $\mathbb{K}^n$) if for every $f \in \mathcal{P}$ and affine function $A : \mathbb{K}^n \to \mathbb{K}^n$ it is the case that $f \circ A \in \mathcal{P}$, where $f \circ A(x) = f(A(x))$.*

Many of the techniques we will explore do extend to a slightly broader class of symmetries, namely "linear-invariance" (invariance under linear transformations of the domain only), but the results, though more general, are more cumbersome to state and so we won't talk about such results here.

The class of affine-invariances is simultaneously "small" and "rich" in the following senses. The number of affine invariances is only quasi-polynomial in the domain size: It is of size $|\mathbb{K}|^{n^2}$ where the domain size is $|\mathbb{K}|^n$, and indeed if $n = O(1)$ it is even polynomial sized. In this sense this is a small collection of invariances (compared to say the setting of statistical properties or graph properties). On the other hand it is still highly symmetric in that it is a "2-transitive" class:

A class of invariances Γ is *t-transitive* if for every pair of t-tuples $(x_1, \ldots, x_t)$ and $(y_1, \ldots, y_t)$ of distinct elements from D, there is a function $\pi \in \Gamma$ satisfying $\pi(x_i) = y_i$ for every $i \in [t]$. 2-transitivity reflects a fairly high level of symmetry and is known to imply "local decodability" (we won't define this concept here) of codes, in the presence of a single local constraint. One of the motivating questions leading to the study of affine-invariance, raised by Alon et al. [1] was whether 2-transitivity of the invariance class is a sufficient condition for local testability. As it turns out the answer is NO, but closely related questions do lead to positive answers. We discuss these below.

5.3 Single-orbit characterizations and Testability

One of the main reasons that a "rich" class of invariances might enable local testing is that the presence of even a single local constraint (as in Definition 2.2) turns into a rich collection of constraints. This motivates our definition of the "orbit" of a constraint, which we define only for affine-invariant properties (though the definition can be generalized).

Definition 5.2. *The orbit of a constraint $C = (\alpha_1, \ldots, \alpha_k; V)$ is the set of constraints* $\mathrm{orbit}(C) = \{A \circ C = (A(\alpha_1), \ldots, A(\alpha_k); V) | A : \mathbb{K}^n \to \mathbb{K}^n \textit{ affine}\}$.

2-transitivity of the affine-invariant class says that the orbit of a single constraint ensures the presence of local constraints relating every pair of function values. Indeed this collection is so rich that in most previously studied affine-invariant properties, the orbit of a single local constraint actually characterized the property.

Definition 5.3. *$\mathcal{P}$ is said to have a k-single orbit characterization if there exists a k-local constraint C such that $f \in \mathcal{P}$ if and only if f satisfies every constraint in* $\mathrm{orbit}(C)$.

One possible hope with affine-invariant properties that satisfy some k-local constraint might be that they also have a $k' = k'(k)$-single orbit characterization, and that k'-single orbit characterized properties are $k'' = k''(k')$-locally testable. If so, this would yield a positive resolution of the question raised in [1]. As it turned out, the second part of this hope did turn out to be true, as captured by the following theorem.

Theorem 5.4 ([33])**.** *If $\mathcal{P}$ is has a k-single orbit characterization, then $\mathcal{P}$ is $(k, \rho/O(k^2), \rho)$-locally testable for every ρ.*

The test used to prove the above theorem is the natural one. Let $C = (\alpha_1, \ldots, \alpha_k; V)$ be a constraint such that $\mathrm{orbit}(C)$ characterizes $\mathcal{P}$. Then the test picks a random constraint in $\mathrm{orbit}(C)$ and accepts f if and only if f satisfies this test. It is easy to see this accepts $f \in \mathcal{P}$ with probability one and the harder part is to see why some function f rejected with low probability is actually close to $\mathcal{P}$. We won't describe the analysis in full, however we will say a few words. The key to the analysis is showing that for every $x \in \mathbb{K}^n$ there is some value $g(x)$ that satisfies the following condition: For almost all affine A satisfying $A(\alpha_1) = x$, it is the case that $(g(x), f(A(\alpha_2), \ldots, f(A(\alpha_k))) \in V$. (I.e., f satisfies the constraint $A \circ C$ if $f(x)$ replaced by $g(x)$). The proof that such a function g exists ends up following, somewhat surprisingly, from properties of tensor products of codes, and this aspect ends up emerging as the technique that unifies many previous results in algebraic property testing. Once the existence of such a function g is established, it is also possible to show

that g satisfies every constraint C and usually g equals f thereby showing f is close to a member (specifically g) of $\mathcal{P}$.

Returning to the broader question, why is this theorem important, it turns out that single-orbit characterizations **seem** to play a very important role in the testing of affine-invariant properties. All known locally testable affine-invariant properties seem to owe their testability to single-orbit characterizability. Most algebraic properties studied early on in property testing (typically low-degree polynomials with different tradeoffs between degrees and field sizes) enjoy the single-orbit property for natural (or well-known) reasons. Functions over high-dimensional vector spaces over small fields also do so for simple reasons. But less obvious cases, like sparse families also end up having single-orbit characterizations which may be somewhat surprising.

Proposition 5.5 (See e.g. [1]). *Let* $\mathrm{RM}(n,r) \subseteq \{\mathbb{F}_2^n \to \mathbb{F}_2\}$ *be the set of evaluations of* n*-variate polynomials of degree at most* r *over* $\mathbb{F}_2$*. Then* $\mathrm{RM}(n.r)$ *has a* 2^{r+1}*-single orbit characterization.*

Note that the locality of the characterization is independent of n, the number of variables. The intuition why such a characterization exists is natural: The basic constraint leading to the above proposition simply examines the given function on some arbitrary $r+1$ dimensional subspace and accepts the function if and only if it agrees with some degree r polynomial on this subspace. The orbit of this constraint now constrains a given function to be of degree at most r on every affine subspace of dimension at most $r+1$. The main step in the analysis (based on basic algebra) shows that every polynomial of degree greater than r actually has degree $r+1$ (maximum possible) on some $r+1$ dimensional affine subspace.

The above is an example of an explicit family that has the single orbit characterization. A somewhat general result, relevant when the domain $\mathbb{K}$ is a small field is the following:

Theorem 5.6 ([33]). *For every* k *and field* $\mathbb{K}$ *extending* $\mathbb{F}$ *there exists a* $k' = k'(k, |\mathbb{K}|)$ *such that for every* n*, every affine-invariant property* $\mathcal{P} \subseteq \{\mathbb{K}^n \to \mathbb{F}\}$ *that satisfies some* k*-local constraint is* k'*-single orbit characterizable.*

The proof of the above theorem follows from some structural analysis of affine-invariant properties that we will describe in the next section. Roughly one can attribute to every property some sort of a "degree bound" describing the highest degree of a function satisfying the property and use this degree to lower bound the size of the local constraint, while using it also to bound from above the size of the single-orbit characterization. We will comment more on such algebraic degrees in later.

Both theorems above are interesting only when we consider functions mapping $\mathbb{K}^n \to \mathbb{F}$ for small $\mathbb{K}$ and large n. We now turn to some results about the case where $n = 1$. This case is the most interesting since all others are special cases of this one, as we explain next.

In what follows we will often find it useful to view a vector space $\mathbb{K}^n$ as a large field $\mathbb{L}$, with $|\mathbb{L}| = |\mathbb{K}^n|$. We will say a bijection $\phi : \mathbb{K}^n \to \mathbb{L}$ is a *natural map* if it preserves the vector space (i.e., $\alpha\phi(x) + \beta\phi(y) = \phi(\alpha x + \beta y)$ for every $\alpha, \beta \in \mathbb{K}$ and $x, y \in \mathbb{K}^n$). Abusing notation somewhat, we say that a property $\mathcal{P} \subseteq \{\mathbb{L} \to \mathbb{F}_2\}$ is a $\mathrm{RM}(n,r)$ property if there exists a natural map $\phi : \mathbb{F}_2^n \to \mathbb{L}$ such that $\mathcal{P}$ is equivalent to $\mathrm{RM}(n,r) \subseteq \{\mathbb{F}_2^n \to \mathbb{F}_2\}$ under ϕ, i.e., $\mathcal{P} = \{f \circ \phi^{-1} | f \in \mathrm{RM}(n,r)\}$.

Proposition 5.7 (Grigorescu [26], Ben-Sasson et al. [6]). *For every* n, r*, if* $\mathcal{P} \subseteq \{\mathbb{F}_{2^n} \to \mathbb{F}_2\}$ *is a* $\mathrm{RM}(n,r)$ *property then* $\mathcal{P}$ *is* 2^{r+1}*-single orbit characterized.*

The above result does not give a new family of properties testable by affine-invariance but it does give a new (and more randomness efficient) test for this property, since the orbit of a constraint is much smaller when the domain of the property is viewed as $\mathbb{F}_{2^n}$, as opposed to $\mathbb{F}_2^n$. (Hope my latex conveys what I mean.)

A much more diverse collection of properties that turn out to have single orbit characterizations are "sparse" properties. The fact that sufficiently sparse properties are testable is something we already covered (in Section 3). However the tests for general sparse properties are quite unstructured. If the property turns out to be affine-invariant one could hope that the test becomes more structured, and this could be shown by showing such properties have single-orbit characterizations. Indeed this hope turned out to be true, as shown by Kaufman and Lovett [31] building on Grigorescu et al. [28]. The first work outlines an approach to getting single-orbit characterizations, but succeeds in a limited setting (for range being $\mathbb{F}_2$, domain being $\mathbb{F}_{2^t}$ for prime t). The second work manages to prove strong technical results that allow the approach to work more broadly, for arbitrary prime fields as range, arbitrary extensions as domain. (They also get some local testability for codes that are quasi-polynomial sized, though not $O(1)$-local testability.) We present their theorem below.

Theorem 5.8 ([31]). *For every prime p and c there exists k such that if $\mathcal{P} \subseteq \{\mathbb{F}_{p^t} \to \mathbb{F}_p\}$ is an affine-invariant property with $|\mathcal{P}| \leq p^{c \cdot t}$, then $\mathcal{P}$ has a k-single orbit characterization.*

While the above results list the "basic" single-orbit characterizable properties, it is possible to get other single-orbit characterized properties by manipulating these basic ones.

Proposition 5.9 (Ben-Sasson et al. [6]). *For every k_1, k_2 there exists t_0 such that for all $t \geq t_0$ the following holds. If $\mathcal{P}_1, \mathcal{P}_2 \subseteq \{\mathbb{F}_{2^t} \to \mathbb{F}_2\}$ are affine invariant properties respectively with k_1- and k_2-single orbit characterizations, then*

1. *$\mathcal{P}_1 \cap \mathcal{P}_2$ has a $(k_1 + k_2)$-single orbit characterization.*
2. *$\mathcal{P}_1 + \mathcal{P}_2 = \{f_1 + f_2 | f_1 \in \mathcal{P}_1, f_2 \in \mathcal{P}_2\}$ has a $(k_1 \cdot k_2)$-single orbit characterization.*

We remark that both parts do require some structural understanding of affine-invariance and single-orbit characterizations. Part (1) follows relatively easily once such understanding is attained while Part (2) seems less immediate and is known only for sufficiently large n (though it is conceivable that it is true for all n).

A different way to get single-orbit characterized properties from known ones is by a "lifting" operator.

Definition 5.10 (Lifting properties). *Let $\mathbb{F} \subseteq \mathbb{K} \subseteq \mathbb{L}$ be fields. Given an affine-invariant property $\mathcal{P} \subseteq \{\mathbb{K} \to \mathbb{F}\}$ its lift $\mathcal{P}' = \textsc{Lift}_{\mathbb{K} \to \mathbb{L}}(\mathcal{P})$ is the affine-invariant property $\mathcal{P}' \subseteq \{\mathbb{L} \to \mathbb{F}\}$ given by $\mathcal{P}' = \{f | \forall \text{ affine } A : \mathbb{L} \to \mathbb{L},\ (f \circ A)|_{\mathbb{K}} \in \mathcal{P}\}$, where for a function $g : \mathbb{L} \to \mathbb{F}$, the function $g|_{\mathbb{K}}$ denotes the restriction of g to the subdomain $\mathbb{K} \subseteq \mathbb{L}$.*

The nice aspect of lifting is that the lift of a k-single orbit characterized property is also k-single orbit characterized. (This follows essentially from the definition, though we admit it may be hard for a first time reader to verify this.) What is more interesting is that lift of sparse properties need not be sparse. So lifting gives new collections of single-orbit characterized properties.

One reason for the lengthy enumeration of single-orbit properties above is that it is actually open whether this enumeration may give a complete explanation of single-orbit characterizability. The following question asks formally if every single orbit characterized property is the sum or intersection of a constant number of lifts of sparse properties or lifts of Reed-Muller properties.

Question 5.11. *Is the following true: For every $k, \mathbb{F}$ there exist c, r, s such that for every $\mathbb{K}$ extending $\mathbb{F}$ and every k-single orbit characterized property $\mathcal{P} \subseteq \{\mathbb{K} \to \mathbb{F}\}$ is derived from at most s properties $\mathcal{P}_1, \ldots, \mathcal{P}_s$, each of which is the lift of a c-sparse property or a Reed-Muller property of degree at most r, by a sequence of sum or intersection operators?*

A complementary question asks how important single-orbit properties are?

Question 5.12. *Is the following true: For every $k, \mathbb{F}$ there exists k' such that for every $\mathbb{K}$ extending $\mathbb{F}$ and every k-locally testable property $\mathcal{P}$ is k'-single orbit characterized?*

If the answer to both questions is affirmative then this would lead to a full characterization of the testability of linear, affine-invariant codes. However, at the moment even the truth of the two statements, leave alone our ability to prove them, seems quite optimistic. Indeed we seem to know relatively little about the structure of affine-invariant properties. In the following sections we report on what they look like, and what we know thus far.

5.4 Structure of affine-invariant properties

In this section we describe some of the "structure" exhibited by affine-invariant properties mapping $\mathbb{K} = \mathbb{F}_{2^n}$ to $\mathbb{F}_2$. The results can be extended to other fields as also to "multivariate functions", but we'll stick to the simpler setting. The study of such results started in [33], and continued in the subsequent works. In the lemmas below we will attempt to point to the place in the literature where the result appears exactly as stated, though possibly slight variants existed earlier.

Fix a property $\mathcal{P} \subseteq \{\mathbb{K} \to \mathbb{F}_2\}$. To explore its structure, we start by recalling that every function from $\mathbb{K}$ to $\mathbb{K}$, and therefore every function from $\mathbb{K}$ to $\mathbb{F}_2$ is a polynomial (of degree at most $2^n - 1$). Much of the structural results will be obtained by looking at the monomials in the support of functions in $\mathcal{P}$. To this end, let us define, for a polynomial $f(x) = \sum_{d=0}^{2^n-1} c_d x^d$, its support to be $\mathrm{supp}(f) = \{d | c_d \neq 0\}$. A central concept associated with an affine-invariant property is its *degree set*, $\mathrm{Deg}(\mathcal{P}) = \cup_{f \in \mathcal{P}} \mathrm{supp}(f)$.

A priori, this set may not seem to be very useful to focus on. Does it really contain much information about $\mathcal{P}$? It turns out it uniquely determines $\mathcal{P}$. To this end, let us define for a set D of integers, $\mathrm{Code}(D) = \{f : \mathbb{K} \to \mathbb{F} | \, \mathrm{supp}(f) \subseteq D\}$. In words, the code of the set D contains all functions with support from D that take on values from $\mathbb{F}_2$. We have the following lemma.

Lemma 5.13 (Ben-Sasson et al. [9])**.** *Let $\mathcal{P} \subseteq \{\mathbb{K} \to \mathbb{F}_2\}$ be affine-invariant. Then $\mathcal{P} = \mathrm{Code}(\mathrm{Deg}(\mathcal{P}))$.*

In other words, any function supported on $\mathrm{Deg}(\mathcal{P})$ is a member of $\mathcal{P}$ with the only restriction being it should be an $\mathbb{F}_2$-valued function. And how restrictive is this condition? Somewhat, but in a very well-understood way. Note that a function $f(x)$ mapping to $\mathbb{F}_2$ should satisfy $f(\alpha)^2 = f(\alpha)$ for every $\alpha \in \mathbb{K}$. In terms of polynomial identities, this implies that $f(x)^2 = f(x) \bmod (x^{2^n} - x)$,

which in turn implies that if $f(x) = \sum_{d=0}^{2^n-1} c_d x^d$, then $c_{2d \bmod (2^n-1)} = c_d^2$ for every d. Among other things this implies that if $d \in \mathrm{Deg}(\mathcal{P})$ then also $2d \in \mathrm{Deg}(\mathcal{P})$. This motivates the notion of the shifts of an integer d or a set of integers S. We let $\mathrm{shift}(d) = \{2^i \cdot d (\mathrm{mod}\ 2^n - 1) | i \in \mathbb{Z}\}$, and let $\mathrm{shift}(S) = \cup_{d \in S} \mathrm{shift}(d)$. We say S is *shift-closed* if $\mathrm{shift}(S) = S$.

We can further ask, what other properties do degree sets satisfy? It turns that the binary representations of integers becomes important in understanding this. For integer d, let $[d]_i$ denote the ith least significant bit in the binary representation of d (so $d = \sum_i [d]_i 2^i$). We say, e is in the *shadow* of d, denoted $e \leq_2 d$, if for all i, $[e]_i \leq [d]_i$. We let the shadow of integer d, denoted $\mathrm{shadow}(d)$, be the set of all integers $e \leq_2 d$, and let $\mathrm{shadow}(S) = \cup_{d \in S} \mathrm{shadow}(d)$. A set S is said to be *shadow-closed* if $\mathrm{shadow}(S) = S$. Shadows and shifts suffice to explain what degree sets of affine-invariant families look like, as formalized below.

Lemma 5.14. *For every $\mathcal{P}$, $\mathrm{Deg}(\mathcal{P})$ is shadow-closed and shift-closed. Conversely if, $S \subseteq \{0, \ldots, 2^n - 1\}$ is a shadow-closed and shift-closed set, then $\mathrm{Code}(S)$ is an affine-invariant family with $S = \mathrm{Deg}(\mathrm{Code}(S))$.*

The key ingredient used in the proofs of the above two lemmas is the identity (an inverse Fourier transform) that for a function $f(x) = \sum_{d=1}^{2^n-1} c_d x^d$, we have $c_d x^d = \sum_{\alpha \in \mathbb{K} - \{0\}} \alpha^{-d} f(\alpha x)$. The reason this is useful in our context is that the right hand side expression is just a linear combination of affine (even linear) transforms of f. This would be a member of $\mathcal{P}$ if the coefficients were from $\mathbb{F}_2$ rather than $\mathbb{K}$. But this is not the case, and so to remedy it, we resort to the trace function given by $\mathrm{Trace}(z) = z + z^2 + z^4 + \cdots + z^{2^{n-1}}$. This is a nice additive function that maps $\mathbb{K}$ to $\mathbb{F}$. When applied to the coefficients $\mathrm{Trace}(\alpha_d)$ on the right in the expression above, it yields the identity $\mathrm{Trace}(c_d x^d) = \sum_{\alpha \in \mathbb{K} - \{0\}} \mathrm{Trace}(\alpha^{-d}) f(\alpha x)$ which is definitely in $\mathcal{P}$ if $f \in \mathcal{P}$. This allows us to separate the monomials occuring in f and treat them essentially separately. With some care, one can show that if $c_d \neq 0$ then $\mathrm{Trace}(\lambda x^d)$ is in $\mathcal{P}$ for every $\lambda \in \mathbb{K}$ and then it is straightforward to argue that $\mathcal{P} = \mathrm{Code}(\mathrm{Deg}(\mathcal{P}))$. The fact that $c_{2d} = c_d^2$ implies the degree sets are shift-closed. And one explores functions of the form $f(x + \alpha) - f(x)$ to show that the degree sets are shadow-closed. The second lemma above is thus easily verified.

Thus looking at the degree sets of affine-invariant properties gives us an alternate, somewhat more explicit, view of affine-invariant properties, but thus far we haven't said anything about what makes a property locally testable. Can we somehow relate local testability to the degrees seen in the degree set? There has been only minimal progress on this front. We explain some of the issues and results next.

5.5 Locality from Structural Properties

First we make a (relatively) simple connection between constraints (or even characterizations) and degree sets. For simplicity we restrict our attention to basic constraints in this section, i.e., constraints of the form $\sum_{i=1}^{k} f(\alpha_i) = 0$.

Lemma 5.15 ([9])**.** *Let $\mathcal{P} \subseteq \{\mathbb{K} \to \mathbb{F}_2\}$ be an affine-invariant property with degree set $S = \mathrm{Deg}(\mathcal{P})$. Then we have:*

1. *$\alpha_1, \ldots, \alpha_k$ form a basic constraint on $\mathcal{P}$ if and only if $\sum_{i=1}^{k} \alpha_i^d = 0$ for all $d \in S$.*

2. *The basic constraint $\alpha_1, \ldots, \alpha_k$ on $\mathcal{P}$ give a k-single orbit characterization of $\mathcal{P}$ if and only if for every $d \notin S$ there exists $e \leq_2 d$ such that $\sum_{i=1}^{k} \alpha_i^e \neq 0$.*

The lemma above follows easily from the structural properties described in the previous section. The importance of this lemma is that it converts questions about existence of constraints/characterizations to questions about the kernel of a "van der Monde" like matrix. Let us introduce this matrix next. For $\vec{\alpha} = (\alpha_1, \ldots, \alpha_k) \in \mathbb{K}$ and set S of integers, let $M = M(\vec{\alpha}, S)$ be the $|S| \times k$ matrix with rows indexed by elements of S and columns by elements of $[k]$ with $M_{d,j} = \alpha_j^d$. If $S = \{0, \ldots, k-1\}$ this would simply be the van der Monde matrix, but our interest is in other S's.

The importance of this matrix to our setting is fairly straightforward. Lemma 5.15 above essentially says the following: $\vec{\alpha}$ is a basic constraint on $\mathcal{P}$ if and only if the all 1's vector is in the right kernel of $M(\vec{\alpha}, \mathrm{Deg}(\mathcal{P}))$ (i.e., $M(\vec{\alpha}, \mathrm{Deg}(\mathcal{P})) \cdot \vec{1} = \vec{0}$. And a somewhat similar statement would also explain when $\vec{\alpha}$ is a characterization. Analyzing conditions when $\vec{1}$ is not in the right kernel turns out to be quite challenging and we have relatively few results of this nature. (This is related to the general theme exploring conditions under which an explicit, non-square, matrix has full column rank. This is an area of relative darkness, especially over finite fields.)

We describe a few limited cases where progress has been made.

Lemma 5.16 (Grigorescu et al. [27])**.** *Let $S = \{1, 2, 4, \ldots, 2^{k-1}\}$ and $\alpha_1, \ldots, \alpha_k \in \mathbb{K}$ be linearly independent over $\mathbb{F}_2$. Then $M(\vec{\alpha}, S)$ is non-singular.*

[27] uses this result to show that there are affine-invariant families that exhibit local constraints (8-local constraints, to be specific) but are not $O(1)$-locally characterized, which resolves the earlier mentioned question raised in [1] negatively.

A more general class of results (incomparable to the above) is given by Ben-Sasson and Sudan [12] who relate the locality of contraints to the "weights" of degrees. To this end, let the weight of integer d, denoted $\mathrm{wt}(d)$, be the number of non-zero bits in the binary representation of d. I.e., $\mathrm{wt}(d) = \sum_i [d]_i$. For set S, we let $\mathrm{wt}(S) = \max_{d \in S}\{\mathrm{wt}(d)\}$. The weight of integers play an important role in understanding affine-invariant properties. For example, we have the following proposition:

Proposition 5.17. *If $\mathcal{P} \subseteq \{\mathbb{K} \to \mathbb{F}_2\}$ is an $\mathrm{RM}(n, r)$ code, then $\mathrm{Deg}(\mathcal{P}) = \{d | \mathrm{wt}(d) \leq r\}$.*

Since degree sets are monotone with respect to inclusion it follows that any property $\mathcal{P}$ with $\mathrm{wt}(\mathrm{Deg}(\mathcal{P})) \leq r$ is contained in $\mathrm{RM}(n, r)$ and thus satisfies a 2^{r+1}-local constraint. The following result gives a weak converse.

Theorem 5.18 ([12])**.** *If $\mathcal{P}$ is an affine invariant property satisfying an k-local constraint, then $\mathrm{wt}(\mathrm{Deg}(\mathcal{P})) < k$.*

Thus the minimum locality of a constraint satisfied by any family is between k and 2^k where $k = 1 + \mathrm{wt}(\mathrm{Deg}(\mathcal{P}))$. Both extremes are known to be tight.

The main lemma leading to the above theorem is yet another rank lower bound for a "generalized van der Monde" matrix.

Lemma 5.19 ([12]). *Let S be a shadow-closed set with $\mathrm{wt}(S) \geq k$. Then $M(\alpha_1, \ldots, \alpha_k, S)$ has rank k.*

Theorem 5.18 may well be a first step towards a potential characterization of $O(1)$-locally testable affine-invariant properties. It provides some restrictions on the scope of locally testable properties, but is still far from pinning them down exactly. We note it provides a useful tool however: For instance, the (current) proof of Proposition 5.9 above uses this theorem as an ingredient.

Finally we mention one more result which is obtained by giving a rank lower bound on a carefully constructed generalized van der Monde matrix. Ben-Sasson et al. [9] construct an example of a locally characterized (LDPC) affine invariant property which is not locally testable, which refutes a more interesting, and weaker, variant of the earlier-mentioned question of Alon et al. [1].

6 Conclusions

In the previous sections we described some general techniques (in that they apply to a wide variety of properties) in the testing of linear properties. Most of these themes developed in an attempt to abstract and generalize some basic property tests including the linearity test from [16] and low-degree tests as in [43, 1, 32, 29]. Still we do not reach the state of the art in the constructions of locally testable codes. To get there we need to capture some more of the techniques in property testing (e.g., those introduced in [2, 39, 40] or in [17, 38]. The former class of results, which focus on testing polynomials, use two aspects of polynomials: (1) The fact that low-degree polynomials are affine-invariant, and (2) the fact that the product of low-degree polynomials is itself low-degree. It is conceivable that one could abstract the latter property and build tests based on these, and it would be interesting to see if it can be done, and if it can suggest new designs of locally testable codes.

On the topic of "invariance", certainly it is our hope that this theme leads to a broad understanding of many themes is property testing (not just in that of testing linear properties). However it is worth stressing that it will still fail to capture many property tests. Indeed it has been shown by Goldreich and Kaufman [22] that random sparse properties show no invariance at all (so this theme has its limitations). In the other direction they also show that invariance, even with local characterizations, is insufficient for testing. ([22] shows this for a non-linear property with some 1-transitive invariance class. The subsequent work of Ben-Sasson et al. [9] mentioned earlier shows this for a linear property that is even affine-invariant.) Despite the limitations, we do believe invariances have significant unifying power (even if it does not explain everything). We also hope that with further understanding it can lead to new classes of locally testable properties with novel additional features if not extremal parameters.

Acknowledgments

First my (ahem) thanks to Lane Hemaspaandra for convincing me to write this survey by imposing an eighteen month deadline. I should learn a trick or two from him! I'd like to thank Eli Ben-Sasson, Oded Goldreich, Elena Grigorescu, Tali Kaufman, Swastik Kopparty, Ghid Maatouk, Or Meir, Dana

Ron, Shubhangi Saraf, Amir Shpilka, and Michael Viderman for their works and (explicit/implicit) opinions that led to this survey. Thanks to Eli and Elena for comments on the earlier draft. This survey would have been even worse without their help.

References

[1] Noga Alon, Tali Kaufman, Michael Krivelevich, Simon Litsyn, and Dana Ron. Testing Reed-Muller codes. *IEEE Transactions on Information Theory*, 51(11):4032–4039, 2005.

[2] Sanjeev Arora and Shmuel Safra. Probabilistic checking of proofs: A new characterization of NP. *Journal of the ACM*, 45(1):70–122, January 1998.

[3] László Babai, Lance Fortnow, and Carsten Lund. Non-deterministic exponential time has two-prover interactive protocols. *Computational Complexity*, 1(1):3–40, 1991.

[4] László Babai, Amir Shpilka, and Daniel Stefankovic. Locally testable cyclic codes. *IEEE Transactions on Information Theory*, 51(8):2849–2858, 2005.

[5] Mihir Bellare, Don Coppersmith, Johan Håstad, Marcos Kiwi, and Madhu Sudan. Linearity testing over characteristic two. *IEEE Transactions on Information Theory*, 42(6):1781–1795, November 1996.

[6] Eli Ben-Sasson, Elena Grigorescu, Ghid Maatouk, Amir Shpilka, and Madhu Sudan. On the sum of single-orbit affine invariant properties. In preparation, 2011.

[7] Eli Ben-Sasson, Venkatesan Guruswami, Tali Kaufman, Madhu Sudan, and Michael Viderman. Locally testable codes require redundant testers. *SIAM Journal on Computing*, 39(7):3230–3247, 2010. Preliminary version appeared in Proc. IEEE CCC 2009.

[8] Eli Ben-Sasson, Prahladh Harsha, and Sofya Raskhodnikova. Some 3CNF properties are hard to test. *SIAM Journal on Computing*, 35(1):1–21, September 2005. Preliminary version in *Proc. STOC 2003*.

[9] Eli Ben-Sasson, Ghid Maatouk, Amir Shpilka, and Madhu Sudan. Symmetric LDPC codes are not necessarily locally testable. *Electronic Colloquium on Computational Complexity (ECCC)*, 17:199, 2010.

[10] Eli Ben-Sasson and Madhu Sudan. Robust locally testable codes and products of codes. *Random Structures and Algorithms*, 28(4):387–402, 2006.

[11] Eli Ben-Sasson and Madhu Sudan. Short PCPs with polylog query complexity. *SIAM J. Comput.*, 38(2):551–607, 2008. Preliminary version in Proc. STOC 2005.

[12] Eli Ben-Sasson and Madhu Sudan. Limits on the rate of locally testable affine-invariant codes. *Electronic Colloquium on Computational Complexity (ECCC)*, 17:108, 2010.

[13] Eli Ben-Sasson and Michael Viderman. Composition of semi-LTCs by two-wise tensor products. In Dinur et al. [18], pages 378–391.

[14] Eli Ben-Sasson and Michael Viderman. Tensor products of weakly smooth codes are robust. *Theory of Computing*, 5(1):239–255, 2009. Preliminary version in Proc. APPROX-RANDOM 2008.

[15] Eli Ben-Sasson and Michael Viderman. Low rate is insufficient for local testability. *Electronic Colloquium on Computational Complexity (ECCC)*, 17:4, 2010.

[16] Manuel Blum, Michael Luby, and Ronitt Rubinfeld. Self-testing/correcting with applications to numerical problems. *Journal of Computer and System Sciences*, 47(3):549–595, 1993.

[17] Irit Dinur. The PCP theorem by gap amplification. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, pages 241–250, New York, 2006. ACM Press. Preliminary version appeared as an ECCC Technical Report TR05-046.

[18] Irit Dinur, Klaus Jansen, Joseph Naor, and José D. P. Rolim, editors. *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, 12th International Workshop, APPROX 2009, and 13th International Workshop, RANDOM 2009, Berkeley, CA, USA, August 21-23, 2009. Proceedings*, volume 5687 of *Lecture Notes in Computer Science*. Springer, 2009.

[19] Irit Dinur, Madhu Sudan, and Avi Wigderson. Robust local testability of tensor products of ldpc codes. In Josep Díaz, Klaus Jansen, José D. P. Rolim, and Uri Zwick, editors, *APPROX-RANDOM*, volume 4110 of *Lecture Notes in Computer Science*, pages 304–315. Springer, 2006.

[20] Oded Goldreich, editor. *Property Testing - Current Research and Surveys [outgrow of a workshop at the Institute for Computer Science (ITCS) at Tsinghua University, January 2010]*, volume 6390 of *Lecture Notes in Computer Science*. Springer, 2010.

[21] Oded Goldreich, Shafi Goldwasser, and Dana Ron. Property testing and its connection to learning and approximation. *JACM*, 45(4):653–750, 1998.

[22] Oded Goldreich and Tali Kaufman. Proximity oblivious testing and the role of invariances. *Electronic Colloquium on Computational Complexity (ECCC)*, 17:58, 2010.

[23] Oded Goldreich and Dana Ron. On proximity oblivious testing. In Michael Mitzenmacher, editor, *STOC*, pages 141–150. ACM, 2009.

[24] Oded Goldreich and Or Sheffet. On the randomness complexity of property testing. In Moses Charikar, Klaus Jansen, Omer Reingold, and José D. P. Rolim, editors, *APPROX-RANDOM*, volume 4627 of *Lecture Notes in Computer Science*, pages 509–524. Springer, 2007.

[25] Oded Goldreich and Madhu Sudan. Locally testable codes and PCPs of almost-linear length. *Journal of the ACM*, 53(4):558–655, 2006. Preliminary version in *FOCS 2002*.

[26] Elena Grigorescu. *Symmetries in Algebraic Property Testing*. PhD thesis, MIT, August 2010.

[27] Elena Grigorescu, Tali Kaufman, and Madhu Sudan. 2-transitivity is insufficient for local testability. In *CCC 2008: Proceedings of the 23rd IEEE Conference on Computational Complexity*, page (to appear). IEEE Computer Society, June 23-26th 2008.

[28] Elena Grigorescu, Tali Kaufman, and Madhu Sudan. Succinct representation of codes with applications to testing. In Dinur et al. [18], pages 534–547.

[29] Charanjit S. Jutla, Anindya C. Patthak, Atri Rudra, and David Zuckerman. Testing low-degree polynomials over prime fields. In *FOCS '04: Proceedings of the Forty-Fifth Annual IEEE Symposium on Foundations of Computer Science*, pages 423–432. IEEE Computer Society, 2004.

[30] Tali Kaufman and Simon Litsyn. Almost orthogonal linear codes are locally testable. In *Proceedings of the Forty-sixth Annual Symposium on Foundations of Computer Science*, pages 317–326, 2005.

[31] Tali Kaufman and Shachar Lovett. Testing of exponentially large codes, by a new extension to Weil bound for character sums. *Electronic Colloquium on Computational Complexity (ECCC)*, 17:65, 2010.

[32] Tali Kaufman and Dana Ron. Testing polynomials over general fields. *SIAM J. Comput.*, 36(3):779–802, 2006.

[33] Tali Kaufman and Madhu Sudan. Algebraic property testing: The role of invariance. Technical Report TR07-111, Electronic Colloquium on Computational Complexity, 2 November 2007. Extended abstract in *Proc. 40th STOC*, 2008.

[34] Tali Kaufman and Madhu Sudan. Sparse random linear codes are locally decodable and testable. In *FOCS*, pages 590–600. IEEE Computer Society, 2007.

[35] Tali Kaufman and Avi Wigderson. Symmetric ldpc codes and local testing. In Andrew Chi-Chih Yao, editor, *ICS*, pages 406–421. Tsinghua University Press, 2010.

[36] Swastik Kopparty and Shubhangi Saraf. Tolerant linearity testing and locally testable codes. In Dinur et al. [18], pages 601–614.

[37] Swastik Kopparty and Shubhangi Saraf. Local list-decoding and testing of random linear codes from high error. In Leonard J. Schulman, editor, *STOC*, pages 417–426. ACM, 2010.

[38] Or Meir. Combinatorial construction of locally testable codes. *SIAM J. Comput.*, 39(2):491–544, 2009.

[39] Alexander Polishchuk and Daniel A. Spielman. Nearly linear-size holographic proofs. In *Proceedings of the Twenty-Sixth Annual ACM Symposium on the Theory of Computing*, pages 194–203, New York, NY, 23-25 May 1994. ACM Press.

[40] Ran Raz and Shmuel Safra. A sub-constant error-probability low-degree test, and a sub-constant error-probability PCP characterization of NP. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, pages 475–484, New York, NY, 1997. ACM Press.

[41] Dana Ron. Algorithmic and analysis techniques in property testing. *Foundations and Trends in Theoretical Computer Science*, 5(2):73–205, 2009.

[42] Ronitt Rubinfeld. Sublinear time algorithms. In *Proceedings of International Congress of Mathematicians*, volume III, pages 1095–1110. European Mathematical Society, 22-30 August 2006.

[43] Ronitt Rubinfeld and Madhu Sudan. Robust characterizations of polynomials with applications to program testing. *SIAM Journal on Computing*, 25(2):252–271, April 1996.

[44] Madhu Sudan. Invariance in property testing. In Goldreich [20], pages 211–227.

[45] Paul Valiant. The tensor product of two codes is not necessarily robustly testable. In Chandra Chekuri, Klaus Jansen, José D. P. Rolim, and Luca Trevisan, editors, *APPROX-RANDOM*, volume 3624 of *Lecture Notes in Computer Science*, pages 472–481. Springer, 2005.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de