Electronic Colloquium on Computational Complexity, Revision 3 of Report No. 80 (2016)

Reducing testing affine spaces to testing linearity of functions

Oded Goldreich

October 22, 2019

Abstract

For any finite field $\mathcal{F}$ and $k < \ell$, we consider the task of testing whether a function $f : \mathcal{F}^\ell \to \{0,1\}$ is the indicator function of an $(\ell - k)$-dimensional affine space. An optimal tester for this property (for the case of $\mathcal{F} = \mathrm{GF}(2)$) was presented by Parnas, Ron, and Samorodnitsky (*SIDMA*, 2002), by mimicking the celebrated linearity tester of Blum, Luby and Rubinfeld (*JCSS*, 1993) and its analysis. We show that the former task (i.e., testing $(\ell - k)$-dimensional affine spaces) can be efficiently reduced to testing the linearity of a related function $g : \mathcal{F}^\ell \to \mathcal{F}^k$. This reduction yields an almost optimal tester for affine spaces (represented by their indicator function).

Recalling that Parnas, Ron, and Samorodnitsky used testing $(\ell-k)$-dimensional affine spaces as the first step in a two-step procedure for testing k-monomials, we also show that the second step in their procedure can be reduced to the special case of $k = 1$.

A preliminary version of this work was posted in April 2016 as a guest column on the *Property Testing Review*. It was significantly revised and appeared as TR16-080 of *ECCC*. The current version is the result of an even more extensive revision. In particular, the reduction of testing affine spaces to testing linearity (of functions) is extended to arbitrary finite fields, many of the technical justifications are elaborated, and some crucial typos are fixed. In addition, the title has been augmented for clarity, the brief introduction has been expanded, and the high level structure has been re-organized (i.e., the original Sections 3 and 5 have been merged and placed after the original Section 4.)

1 Introduction

Property Testing is the study of super-fast (randomized) algorithms for approximate decision making. These algorithms are given direct access to items of a huge data set, and determine whether this data set has some predetermined (global) property or is far from having this property, while accessing a small portion of the data set. Thus, property testing is a relaxation of decision problems and it focuses on algorithms, called *testers*, that only read parts of the input. Consequently, the testers are modeled as oracle machines and the inputs are modeled as functions to which the tester has an oracle access.

This paper refers to several basic tasks in property testing, including testing linearity, testing dictatorship, testing (monotone) k-monomials, and testing affine spaces. Whereas the first three tasks refer explicitly to an input function (i.e., the object they test is naturally viewed as a function), in the case of testing affine spaces the object is a set of points and representing this set by an indicator function is a natural choice but not an immediate one. In particular, this is a very redundant representation in the case that the set is sparse, but we will consider the case of relatively dense sets. Furthermore, this representation arises naturally in the study of testing k-monomials.

ISSN 1433-8092

The problem of testing whether a Boolean function is a (monotone) k-monomial was first studied by Parnas, Ron, and Samorodnitsky [6]. The tester that they presented generalizes their tester of dictatorship (i.e., the case $k = 1$), and does so by following the same two-step strategy and using similar arguments at each step. This raises the question of whether the case of general k can be reduced to the special case of $k = 1$. (This question occurred to me when writing [4, Sec. 5.2.2], and the first version of the current paper was written at that time.)

Specifically, the first step in the strategy of Parnas, Ron, and Samorodnitsky [6] is *testing whether the input function $f : \{0,1\}^\ell \to \{0,1\}$ describes an $(\ell - k)$-dimensional affine space*, where the space described by f is $f^{-1}(1)$. In the case of dictatorship (i.e., $k = 1$), this amounts to testing whether f itself is an affine function, but in the case of $k > 1$ a more general task arises (i.e., testing whether $f^{-1}(1)$ is an $(\ell - k)$-dimensional affine space is fundamentally different from testing whether f is affine). In the second step, one tests whether this affine space is of the right form (i.e., is a translation by 1^ℓ of a linear space spanned by axis-parallel vectors). In the case of $k = 1$, the latter task amounts to testing whether the affine function depends on a single variable, but in the case of $k > 1$ another more general task arises.

Both these general tasks were solved by Parnas, Ron, and Samorodnitsky [6], but their solutions mimic the solutions used in the case of $k = 1$. Furthermore, in both cases, the generalization is very cumbersome. We find this state of affairs quite annoying, and believe that it is more appealing to reduce the general case to the special case.

Our contribution. This paper partially achieves this goal by (1) replacing the first step of [6] with a reduction to the (extensively studied) problem of testing linearity of functions, and (2) *reducing the second step to its special case of $k = 1$*. Specifically, we first *reduce the problem of testing affine spaces to the problem of testing the linearity of functions.* This reduction actually hold over any finite field, whereas the application to testing monomials only uses the case of the binary field (which is the case treated in [6]). The complexity of the testers that we deriver is only slightly inferior to the complexity of the corresponding optimal testers of [6]; specifically, in the relevant case of $\epsilon = O(2^{-k})$, we get a complexity bound of $\widetilde{O}(\log(1/\epsilon) \cdot 2^k + O(1/\epsilon)$ rather than $O(1/\epsilon)$. (Indeed, the bounds coincide for $\epsilon < 2^{-k}/\widetilde{O}(k)$.)

Organization. In Section 2 we recall the standard definition of property testing and formally define the properties considered in this paper. The reduction of testing affine spaces to testing linearity of functions is presented in Section 3, whereas the problem of testing monomials is considered in Section 4.

2 Preliminaries

We assume that the reader is familiar with the basic definition of property testing (see, e.g., [4]), but for sake of good order we reproduce it here. The basic definition refers to functions with domain D and range R.

Definition 2.1 (a tester for property Π): *Let Π be a set of functions of the form $f : D \to R$. A* tester for Π *is a probabilistic oracle machine, denoted T, that, on input a* proximity parameter ϵ *and oracle access to a function $f : D \to R$, outputs a binary verdict that satisfies the following two conditions.*

1. T accepts inputs in Π: *For every $\epsilon > 0$, and for every $f \in \Pi$, it holds that* $\mathbf{Pr}[T^f(\epsilon) = 1] \geq 2/3$.

2. T rejects inputs that are ϵ-far from Π: *For every $\epsilon > 0$, and for every function $f : D \to R$ that is ϵ-far from Π it holds that $\mathbf{Pr}[T^f(\epsilon) = 0] \geq 2/3$, where f is* ϵ-far from Π *if for every $g \in \Pi$ it holds that $|\{x \in D : f(x) \neq g(x)\}| > \epsilon \cdot |D|$.*

If the first condition holds with probability 1 (i.e., $\mathbf{Pr}[T^f(\epsilon) = 1] = 1$)*, then we say that T has* one-sided error*; otherwise, we say that T has* two-sided error*.*

We focus on the query complexity of such testers, while viewing $|D|$ as an additional parameter. We seek testers of query complexity that is independent of $|D|$, which means that the complexity will be a function of the proximity parameter ϵ and an auxiliary parameter k (of the two properties that we consider).

The properties we shall consider refer to functions over the domain $\mathcal{F}^\ell$, where $\mathcal{F}$ is a finite field. (In the previous versions of this paper, we confined ourselves to the case that $\mathcal{F}$ is the two-element field GF(2), which is the case treated in [6].)

Definition 2.2 (affine spaces): *For fixed $k, \ell \in \mathbb{N}$ and a finite field $\mathcal{F}$, we say that the function $f : \mathcal{F}^\ell \to \{0,1\}$* describes an $(\ell - k)$-dimensional affine space *if $f^{-1}(1) = \{x \in \mathcal{F}^\ell : f(x) = 1\}$ is an $(\ell - k)$-dimensional affine space; that is, $f^{-1}(1) = \{yG + s : y \in \mathcal{F}^{\ell-k}\}$, where $G \in \mathcal{F}^{(\ell-k)\times\ell}$ is an $(\ell - k)$-by-ℓ full-rank matrix and $s \in \mathcal{F}^\ell$. When $s = 0^\ell$, the* described space is linear*.*

Definition 2.3 (linear functions): *For fixed $k, \ell \in \mathbb{N}$ and a finite field $\mathcal{F}$, we say that $g : \mathcal{F}^\ell \to \mathcal{F}^k$ is* linear *if $g(x + y) = g(x) + g(y)$ for all $x, y \in \mathcal{F}^\ell$. Equivalently, $g(z) = zT$ for a ℓ-by-k matrix T. We say that f is* affine *if $f(z) = f'(z) + s$ for a linear function f' and some $s \in \mathcal{F}^k$.*

When $k = 1$ and $\mathcal{F} = \mathrm{GF}(2) \equiv \{0,1\}$, it holds that $f : \mathcal{F}^\ell \to \{0,1\}$ describes an $(\ell - k)$-dimensional affine space (resp., linear space) if and only if f is a non-constant affine function (resp., $f + 1$ is a non-constant linear function). But in the other cases this does not hold; in particular, for other fields a non-constant affine function must range over $\mathcal{F}$ rather than over $\{0,1\}$, whereas for $\mathcal{F} = \mathrm{GF}(2)$ and $k > 1$ the densities do not match (i.e., an $(\ell - k)$-dimensional affine space over GF(2) has density 2^{-k} but $f^{-1}(1)$ has density 1/2, for any non-constant affine function $f : \mathrm{GF}(2)^\ell \to \mathrm{GF}(2)$).

Conventions. When writing $\mathbf{Pr}_x[\mathrm{event}(x)]$ we refer to the case that x is selected uniformly in a set that is clear from the context; we sometimes spell out this set by writing $\mathbf{Pr}_{x\in S}[\mathrm{event}(x)]$. For sake of simplicity, we often use the phrase "with high probability" (abbrev., "w.h.p."), which mean that we can obtain arbitrary high constant probability smaller 1 (e.g., 0.99). The image of a function $f : D \to R$ is the set $\{f(e) : e \in D\} \subseteq R$. The symbol $\perp$ denotes a special symbol that is not in $\mathcal{F}^k$. When stating time complexity bounds we shall assume that basic operations on elements of $\mathcal{F}^\ell$ have unit cost.

3 The reduction (of testing affine spaces to testing linearity)

We start by restating the problem. We are given access to a function $h : \mathcal{F}^\ell \to \{0,1\}$ and wish to test whether $h^{-1}(1)$ is an $(\ell - k)$-dimensional affine subspace by reducing this problem to testing linearity (of a function). We present two reductions. The first (and simpler) reduction increases the complexities by a factor of $|\mathcal{F}|^k$, whereas the second reduction only incurs an overhead of $\widetilde{O}(\log(1/\epsilon))$. The first reduction will be used as a subroutine in the second one, and it also provides a good warm-up towards the second one.

3.1 Simplifying assumptions

First, note that we may assume that $\epsilon = O(|\mathcal{F}|^{-k})$, which means that $|\mathcal{F}|^k = O(1/\epsilon)$, since the case of $\epsilon > 4 \cdot |\mathcal{F}|^{-k}$ can be handled by estimating the density of $h^{-1}(1)$ (and accepting iff the estimate is below $\epsilon/2$). Specifically, note that any function that describes an $(\ell - k)$-dimensional subspace (over $\mathcal{F}$) is at distnace exactly $|\mathcal{F}|^{-k}$ from the all-zero function. Hence, if h is 0.75ϵ-close to the all-zero function, then it is ϵ-close to describing a $(\ell - k)$-dimensional subspace, since $|\mathcal{F}|^{-k} < \epsilon/4$, and it is OK to accept it. On the other hand, if h is 0.25ϵ-far from the all-zero function, then it cannot describe a $(\ell - k)$-dimensional subspace (again, since $\epsilon/4 > |\mathcal{F}|^{-k}$), and it is OK to reject it. Hence, we have

Claim 3.1 (reducing to the case of $\epsilon \leq 4 \cdot |\mathcal{F}|^{-k}$): *Testing whether a function $h : \mathcal{F}^\ell \to \{0,1\}$ describes a $(\ell - k)$-dimensional affine subspace can be randomly reduced to the case of $\epsilon \leq 4 \cdot |\mathcal{F}|^{-k}$, where the reduction introduces an additive overhead of $O(|\mathcal{F}|^k)$ queries.*

(An alternative justification boils down to resetting the proximity parameter to $\min(\epsilon, 4 \cdot |\mathcal{F}|^{-k})$; that is, on input proximity parameter ϵ and oracle h, we invoke the given tester on proximity parameter $\min(\epsilon, 4 \cdot |\mathcal{F}|^{-k})$ and provide it with oracle access to h.)

Another simplifying assumption is that we are dealing with linear subspaces rather than with affine ones. Actually, we present a reduction of the general case to this special case.[1]

Claim 3.2 (reducing to the linear case): *Testing whether a function $h : \mathcal{F}^\ell \to \{0,1\}$ describes a $(\ell - k)$-dimensional affine subspace can be randomly reduced to testing whether a function $h' : \mathcal{F}^\ell \to \{0,1\}$ describes a $(\ell - k)$-dimensional linear subspace, where the reduction introduces an additive overhead of $O(|\mathcal{F}|^k)$ queries.*

The foregoing randomized reduction has a two-sided error probability; obtaining an analogous one-sided error reduction is left as an open problem. Yet, since the known testers for linear subspaces (i.e., of [6] and of this paper) have two-sided error probability, our use of Claim 3.2 cause no real loss.

Proof: On input parameter $\epsilon > 0$ and oracle access to h, we proceed as follows.

1. Select uniformly a sample of $O(|\mathcal{F}|^k)$ points in $\mathcal{F}^\ell$. If h evaluates to 0 on all these points, then reject. Otherwise, let u be a point in this sample such that $h(u) = 1$.
2. Invoke the tester for linear subspaces on input parameter ϵ and oracle access to h' defined by $h'(x) \stackrel{\text{def}}{=} h(x + u)$, and output its verdict. That is, each query x to h' is emulated by making the query $x + u$ to h.

The overhead of the reduction is due to Step 1, whereas in Step 2 we just invoke the tester for the special case.

If h describes an $(\ell - k)$-dimensional affine subspace, then, with high probability, Step 1 finds $u \in h^{-1}(1)$, since $h^{-1}(1)$ has density $|\mathcal{F}|^{-k}$, and we proceed to Step 2. But in this case it holds that $h'(x) = 1$ if and only if $x + u \in h^{-1}(1)$, which means that h' describes the $(\ell - k)$-dimensional linear space $h^{-1}(1) - u$, and so the invoked test accepts (w.h.p.).

(Indeed, if $H = \{yG+s : y \in \mathcal{F}^{\ell-k}\}$ is an affine space (as in Definition 2.2) and $u = (zG+s) \in H$, then $H - u = \{yG + s - u : y \in \mathcal{F}\} = \{yG + s - (zG + s) : y \in \mathcal{F}^{\ell-k}\} = \{(y - z)G : y \in \mathcal{F}^{\ell-k}\}$ is a

[1] This reduction somewhat simplifies the presentation in Section 3.2, and more significantly so in Section 3.3.

linear space. Likewise, if $H' = \{yG : y \in \mathcal{F}^{\ell-k}\}$ is a linear space, then $H + u = \{yG + u : y \in \mathcal{F}\}$ is an affine space.)

On the other hand, if h is ϵ-far from being an $(\ell - k)$-dimensional affine subspace, then either Step 1 rejects or else $u \in h^{-1}(1)$. But in this case h' (which was defined by $h'(x) \stackrel{\rm def}{=} h(x + u)$) must be ϵ-far from describing an $(\ell - k)$-dimensional linear subspace. This is so because if h' is ϵ-close to g' that describes an $(\ell - k)$-dimensional linear subspace (i.e., g' describes the linear space $\{yG : y \in \mathcal{F}^{\ell-k}\}$), then $g(x) \stackrel{\rm def}{=} g'(x - u)$ (equiv., $g(x + u) = g'(x)$) describes an affine space (i.e., g describes the affine space $\{yG + u : y \in \mathcal{F}^{\ell-k}\}$), whereas h is ϵ-close to g (since $h(x) = h'(x - u)$). Hence, in this case (i.e., h' is ϵ-far from describing an $(\ell - k)$-dimensional linear subspace), Step 2 rejects with high probability. ■

3.2 The first reduction

The key step is the definition of a function $g : \mathcal{F}^\ell \to \mathcal{F}^k \cup \{\bot\}$ such that if $H \stackrel{\rm def}{=} h^{-1}(1)$ is an $(\ell-k)$-dimensional linear space, then g is linear (with image $\mathcal{F}^k$) and $g^{-1}(0^k) = H$. Furthermore, in that case, $g(x)$ indicates one of the $|\mathcal{F}|^k$ translations of H in which x resides; that is, if $v^{(1)}, ..., v^{(k)} \in \mathcal{F}^\ell$ form a basis for the k-dimensional space that complements H, then $g(x)$ represents coefficients $(c_1, ..., c_k) \in \mathcal{F}^k$ such that $x \in H - \sum_{i\in[k]} c_i v^{(i)}$.

Indeed, the definition of g is based on any fixed sequence of linearly independent vectors $v^{(1)}, ..., v^{(k)} \in \mathcal{F}^\ell$ such that for every non-zero sequence of coefficients $(c_1, ..., c_k) \in \mathcal{F}^k$ it holds that $\sum_{i\in[k]} c_i v^{(i)} \notin H$. Such sequences of vectors exist[2] if H is an $(\ell - k)$-dimensional linear space, and so we can find such a sequence in this case (by random sampling and querying h). Failure to find such a sequence will provide good justification for ruling that H is not an $(\ell - k)$-dimensional linear space.

Fixing such a sequence of $v^{(i)}$'s, we define $g : \mathcal{F}^\ell \to \mathcal{F}^k \cup \{\bot\}$ such that $g(x) = (c_1, ..., c_k)$ if $(c_1, ..., c_k) \in \mathcal{F}^k$ is the unique sequence that satisfies $x + \sum_{i\in[k]} c_i v^{(i)} \in H$ and let $g(x) = \bot \notin \mathcal{F}^k$ otherwise. Indeed, a unique sequence $(c_1, ..., c_k) \in \mathcal{F}^k$ exists for each $x \in \mathcal{F}^\ell$ if H is an $(\ell - k)$-dimensional linear space, and in that case $g(x) \in \mathcal{F}^k$ for every $x \in \mathcal{F}^\ell$. But when H is not an $(\ell - k)$-dimensional linear space, it may happen that for some (or even all) x's there is no sequence $(c_1, ..., c_k) \in \mathcal{F}^k$ such that $x + \sum_{i\in[k]} c_i v^{(i)} \in H$; similarly, it may happen that there are several different sequences $(c_1, ..., c_k) \in \mathcal{F}^k$ that satisfy $x + \sum_{i\in[k]} c_i v^{(i)} \in H$. Anyhow, using matrix notation, we restate the foregoing definition next (where the $v^{(i)}$'s are the rows of the matrix V).

Definition 3.3 (the function $g = g_{H,V}$): *Let V be a k-by-ℓ full-rank matrix over $\mathcal{F}$ such that $cV \in H$ implies $c = 0^k$. Then, $g_{H,V} : \mathcal{F}^\ell \to \mathcal{F}^k \cup \{\bot\}$ is defined such that $g_{H,V}(x) = c$ if $c \in \mathcal{F}^k$ is the unique vector that satisfies $x + cV \in H$, and $g_{H,V}(x) = \bot$ if the number of such k-long vectors is not one.*

(Whenever we say that g is linear, we mean, in particular, that it never assumes the value $\bot$.)[3]

Claim 3.4 (H versus $g_{V,H}$): *Let V be as in Definition 3.3. Then, H is an $(\ell - k)$-dimensional linear space if and only if $g = g_{H,V}$ is a linear function with image $\mathcal{F}^k$.*

It follows that if g is ϵ-close to being a linear function with image $\mathcal{F}^k$, then $g^{-1}(0^k)$ is ϵ-close to being an $((\ell - k)$-dimensional) linear space (i.e., the indicator functions of these sets are ϵ-close).

[2] Actually, the density of suitable k-long sequences in H is $\prod_{i\in[k]}(1 - |\mathcal{F}|^{-i}) > 1/4$.

[3] Indeed, when emulating g for the linearity tester, we shall reject if we ever encounter the value $\bot$.

To see this, consider a linear g' that is ϵ-close to g, and note that $H' = \{x \in \mathcal{F}^\ell : g'(x) = 0^k\}$ is ϵ-close to $g^{-1}(0^k)$ (equiv., the Boolean function that indicates membership in H' is ϵ-close to h).

Proof: First note that $g^{-1}(0^k) \subseteq H$ always holds, since $g(x) = c$ implies $x + cV \in H$ (and so $g(x) = 0^\ell$ implies $x \in H$). Furthermore, equality (i.e., $g^{-1}(0^k) = H$) holds if g never assumes the value $\perp$, since in this case $x + cV \in H$ implies that $g(x) = c$ (and so $x \in H$ implies $g(x) = 0^\ell$).

Now, on the one hand, if g is a linear function with image $\mathcal{F}^k$ (i.e., $g(x) = xT$ for some full-rank ℓ-by-k matrix T), then $H = g^{-1}(0^k)$ (i.e., $H = \{x \in \mathcal{F}^\ell : xT = 0^k\}$), which implies that H is an $(\ell - k)$-dimensional linear subspace (since $H = \{yG : y \in \mathcal{F}^{\ell-k}\}$ for any G that is a basis of the space orthogonal to $T^\top$).[4]

On the other hand, if H is an $(\ell - k)$-dimensional linear space, then, for some full-rank $(\ell - k)$-by-ℓ matrix G, it holds that $H = \{yG : y \in \mathcal{F}^{\ell-k}\}$. In this case, for every $x \in \mathcal{F}^\ell$ there exists a *unique* representation of x as $yG - cV$, since V is a basis for a k-dimensional linear space that complements the $(\ell - k)$-dimensional linear space H, which implies $x + cV = yG \in H$, and so $g(x) = c$. It follows that the image of g equals $\mathcal{F}^k$ (since $g(cV) = c$ for every $c \in \mathcal{F}^k$) and that g is linear, since for every $x = yG - cV$ and $x' = y'G - c'V$ in $\mathcal{F}^\ell$, it holds that $g(x) + g(x') = c + c'$ and $c + c' = g(y''G - (c + c')V)$ for every $y'' \in \mathcal{F}^{\ell-k}$ (and in particular for $y'' = y + y'$, which implies that $c + c' = g(x + x')$). ■

Claim 3.5 (finding V): *If $H = h^{-1}(0^k)$ is an $(\ell - k)$-dimensional linear space, then a matrix V as underlying the definition of g can be found* (w.h.p.) *by making $O(|\mathcal{F}|^k)$ queries to h.*

Proof: The matrix V can be found in k iterations as follows. In the i^{th} iteration we try to find a vector $v^{(i)}$ such that $\sum_{j\in[i]} c_j v^{(j)} \notin H$ holds for every $(c_1, ..., c_i) \in \mathcal{F}^i \setminus \{0^i\}$. In each trial, we pick $v^{(i)}$ at random, noting that the probability of success is $1 - |\mathcal{F}|^{i-1} \cdot |\mathcal{F}|^{-k} \geq 1/2$, whereas the condition can be checked by making $|\mathcal{F}|^i - 1$ queries to h. (Actually, $|\mathcal{F}|^{i-1}$ queries suffice for checking in the i^{th} iteration, since it suffuices to check the cases in which $c_i = 1$.) ■

Combining the above two claims, the desired reduction follows (as detailed next). Note that this reduction has two-sided error, and that the resulting tester has query complexity $O(|\mathcal{F}|^k/\epsilon)$ (rather than $O(1/\epsilon)$, all in case $\epsilon < 4 \cdot |\mathcal{F}|^{-k}$).[5] Recall that we have already justified the assumption $\epsilon = O(|\mathcal{F}|^{-k})$ (see Claim 3.1). In fact, we are going to assume that $\epsilon \leq 0.1/t$ for some $t = O(|\mathcal{F}|^k)$, by possibly resetting $\epsilon \leftarrow \min(\epsilon, 0.1/t)$.

Algorithm 3.6 (testing whether H is an $(\ell-k)$-dimensional linear space): *For a universal constant γ, on input a proximity parameter $\epsilon \in (0, \gamma^{-1} \cdot |\mathcal{F}|^{-k})$ and oracle access to $h : \mathcal{F}^\ell \to \{0, 1\}$, specifying $H = h^{-1}(1)$, proceed as follows.*

1. Find an adequate matrix V: *Using $O(|\mathcal{F}|^k)$ queries to h, try to find a k-by-ℓ full-rank matrix V such that for any non-zero $c \in \mathcal{F}^k$ it holds that $cV \notin H$. If such a matrix V is found, then proceed to the next step. Otherwise, reject.*

[4] Alternatively, if $g(x + x') = g(x) + g(x')$ for every $x, x' \in \mathcal{F}^\ell$, then $x, x' \in H$ implies $x + x' \in H$ (for every $x, x' \in \mathcal{F}^\ell$), since $g(x) = g(x') = 0^k$ implies $g(x + x') = 0^k$. Hence, $H = g^{-1}(0^k)$ is a linear subspace. Lastly, we note that this subspace has dimension $\ell - k$, since the image of g equals $\mathcal{F}^k$ and $|g^{-1}(0^k)| = |g^{-1}(c)|$ holds for every $c \in \mathcal{F}^k$.

[5] Needless to say, we would welcome a one-sided error reduction. Recall that the case $\epsilon \geq 4 \cdot |\mathcal{F}|^{-k}$ can be handled by density estimation. A complexity improvement for the main case (of $\epsilon < 4 \cdot |\mathcal{F}|^{-k}$) appears in Section 3.3.

2. Test whether the function $g = g_{H,V}$ is linear: *Invoke a linearity test with proximity parameter ϵ, while providing it with oracle access to the function $g = g_{H,V}$. When the tester queries g at x, query h on $x + cV$ for all $c \in \mathcal{F}^k$, and answer accordingly; that is, the answer is c if c is the unique vector satisfying $h(x + cV) = 1$, otherwise* (i.e., $g(x) = \perp$) *the execution is suspended and the algorithm rejects.*

   If the linearity tester accepts, then proceed to the next step. Otherwise, reject.

3. Check whether the image of g equals $\mathcal{F}^k$: *Assuming that g is ϵ-close to linear, check whether the image of g equals $\mathcal{F}^k$ as follows.*

   (a) *Select uniformly at random a target $c \in \mathcal{F}^k$.*

   (b) *Select uniformly at random a sample S of $t = \gamma \cdot |\mathcal{F}|^k$ elements in $\mathcal{F}^\ell$, and accept if and only if there exists $x \in S$ such that $x + cV \in H$* (i.e., $h(x + cV) = 1$).

   We stress that we do not compute g at each $x \in S$, which would have required $|\mathcal{F}|^k$ queries to h per each x, but rather check whether $h(x+cV) = 1$ by making a single query to h.

   (Recall that if g is linear and its image equals $\mathcal{F}^k$, then each $c \in \mathcal{F}^k$ has $|\mathcal{F}|^{\ell-k}$ pre-images under g, and (w.h.p.) S contains such a pre-image. On the other hand if g is linear and its image is a proper subsets of $\mathcal{F}^k$, then at most $|\mathcal{F}|^{k-1}$ of the elements of $\mathcal{F}^k$ have a pre-image under g.)

Recalling that linearity testing (with proximity parameter ϵ) has complexity $O(1/\epsilon)$, the complexity of the foregoing algorithm is $O(|\mathcal{F}|^k) + O(1/\epsilon) \cdot |\mathcal{F}|^k + O(|\mathcal{F}|^k)$, where the three terms correspond to the three steps.

Proposition 3.7 (analysis of Algorithm 3.6): *Assuming that $\epsilon \leq 0.1/t$, the following holds.*

1. *Algorithm 3.6 accepts every function h that describes an $(\ell - k)$-dimensional linear space with high probability.*

2. *Algorithm 3.6 rejects every function h that is ϵ-far from describing an $(\ell - k)$-dimensional linear space with probability at least* 0.4.

Repeating Algorithm 3.6 for a constant number of times, we obtain a tester for $(\ell - k)$-dimensional linear spaces over $\mathcal{F}$ with query complexity $O(|\mathcal{F}|^k/\epsilon)$.

Proof: Suppose that H is an $(\ell - k)$-dimensional linear space. Then, by Claim 3.5, with high probability (i.e., with probability $1 - (1 - |\mathcal{F}|^{-k})^{O(|\mathcal{F}|^k)} > 0.99$), a suitable matrix V will be found (in Step 1) and Step 2 will accept, since (by Claim 3.4) the function $g_{H,V}$ is linear and its image equals $\mathcal{F}^k$. Likewise, Step 3 will accept with high probability (i.e., with probability $1 - (1 - |\mathcal{F}|^{-k})^{O(|\mathcal{F}|^k)} > 0.99$).

On the other hand, if h is ϵ-far from describing an $(\ell - k)$-dimensional linear space, then either no suitable matrix V is found in Step 1 or $g_{H,V}$ is ϵ-far from the set of linear functions with image $\mathcal{F}^k$ (see discussion following Claim 3.4). Now, if $g_{H,V}$ is ϵ-far from being linear, then with high probability Step 2 will reject. Otherwise, $g_{H,V}$ is ϵ-close to a linear function g' that has an image that is a proper subject of $\mathcal{F}^k$, and in this case Step 3 will reject with probability at least $1 - {}^{-1} - \mathtt{t} \cdot \epsilon$, since c falls outside the image of g' with probability at least $1 - |\mathcal{F}|^{-1}$, and with probability $1 - t \cdot \epsilon$ all queries to $g_{H,V}$ are answered in agreement with g'. Using $\epsilon \leq 0.1/t$, the claim follows. ■

A modified version of Algorithm 3.6 that works also for $\epsilon = \Omega(1)$. In light of Claim 3.1, we may assume that $\epsilon \leq 0.1/t$ as done above. Note that $t = \Omega(|\mathcal{F}|^k)$ must hold in order to guarantee the acceptance of functions having the property (in Step 3), whereas the hypothesis $\epsilon \leq 0.1/t$ was used only in the analysis of Step 3 (where functions that are ϵ-close to a linear function with an image that is a proper subject of $\mathcal{F}^k$ were shown to be rejected with probability at least $0.5 - t \cdot \epsilon$). However, towards presenting the more efficient reduction (in Section 3.3), it is useful to consider the task performed in Step 3 also in the case that $\epsilon = \Omega(1)$. Actually, we shall consider a modification of Algorithm 3.6 in which Step 3 is modified, whereas the other steps remain unchanged.

Algorithm 3.8 (Algorithm 3.6, modified): *On input a proximity parameter $\epsilon \in (0,1)$ and oracle access to $h : \mathcal{F}^\ell \to \{0,1\}$, specifying $H = h^{-1}(1)$, perform Steps 1 and 2 as in Algorithm 3.6, and then* test whether the image of g equals $\mathcal{F}^k$ *as follows.*

(Step 3, modified): *Select uniformly at random a target $c \in \mathcal{F}^k$, and a sample S of $t = O(|\mathcal{F}|^k)$ elements in $\mathcal{F}^\ell$, as in the original Step 3. If S contains some x such that $h(x+cV) = 1$ holds, then pick such an x arbitrarily and accept if and only if c is the only vector c' that satisfies $h(x + c'V) = 1$.*

(Indeed, this augmentation of the original Step 3 requires $|\mathcal{F}|^k - 1$ additional queries. We stress that we perform the check of uniqueness only for one $x \in S$ that satisfies $h(x+cV) = 1$.)

(As in the original Step 3, the list of x's that satisfy $h(x + cV) = 1$ is determined by querying h on $x + cV$ for each $x \in S$.)

We show that for a sufficiently small constant $\epsilon_0 > 0$ (e.g., $\epsilon_0 = |\mathcal{F}|^k/10t = \Omega(1)$), Algorithm 3.8 is essentially a tester of $(\ell - k)$-dimensional linear spaces.

Proposition 3.9 (analysis of Algorithm 3.8): *Let $\epsilon_0 \leq |\mathcal{F}|^k/10t$. For any $\epsilon > \epsilon_0$, the following holds.*

1. *Algorithm 3.8 accepts every function h that describes an $(\ell - k)$-dimensional linear space with probability at least* 0.9.

2. *Algorithm 3.8 rejects every function h that is ϵ-far from describing an $(\ell - k)$-dimensional linear space with probability at least* 0.4.

Again, by standard error reduction and resetting $\epsilon \leftarrow \min(\epsilon, \epsilon_0)$, we obtain a tester for $(\ell - k)$-dimensional linear spaces over $\mathcal{F}$.

Proof: We focus on the analysis of (the modified) Step 3, since the analysis of the other steps provided before did not depend on the hypothesis $\epsilon < 0.1/t$. We first observe that if h describes an $(\ell - k)$-dimensional linear space, then g is linear with image $\mathcal{F}^k$, and Step 3 accepts with probability at least 0.99, since $\mathbf{Pr}_S[S \cap H = \emptyset] < 0.01$. Hence, we turn to the case that h is ϵ-far from describing an $(\ell - k)$-dimensional linear space, and focus on the case that g is ϵ-close to a linear function that has an image that is a proper subset of $\mathcal{F}^k$.

For each $c \in \mathcal{F}^\ell$, let W_c denote the set of x's that satisfy the condition checked by the modified Step 3 (i.e., $x \in W_c$ iff c is the unique k-long vector that satisfies $h(x+cV) = 1$). Note that $x \in W_c$ if and only if $g_{H,V}(x) = c$. Now, given that $g = g_{H,V}$ is ϵ_0-close to a linear function g' with image that is a proper subset of $\mathcal{F}^k$, there exists a set $B \subseteq \mathcal{F}^k$ of size at least $|\mathcal{F}|^k - |\mathcal{F}|^{k-1}$ (i.e., the vectors that are not in the image of g') such that $\sum_{c \in B} |W_c| \leq \epsilon_0 \cdot |\mathcal{F}|^\ell$, since $\bigcup_{c \in B} W_c = \bigcup_{c \in B} \{x : g(x) = c\}$ must be contained in $\{x \in \mathcal{F}^\ell : g(x) \neq g'(x)\}$, whereas the W_c's are disjoint. It follows that, in this case (i.e., $g = g_{H,V}$ is ϵ_0-close to a linear function g' with image that is a proper subset of $\mathcal{F}^k$), Step 3 rejects with probability at least

$$\begin{aligned}
\mathbf{Pr}_c[c \in B] \cdot \mathbf{Pr}_{c,S}[S \cap W_c = \emptyset | c \in B] &= \frac{|B|}{|\mathcal{F}|^k} \cdot \frac{1}{|B|} \cdot \sum_{c \in B} \mathbf{Pr}_{x \in \mathcal{F}^\ell}[x \notin W_c]^t \\
&= \frac{1}{|\mathcal{F}|^k} \cdot \sum_{c \in B} (1 - \mathbf{Pr}_{x \in \mathcal{F}^\ell}[x \in W_c])^t \\
&\geq \frac{|B|}{|\mathcal{F}|^k} - \frac{1}{|\mathcal{F}|^k} \cdot \sum_{c \in B} \frac{t \cdot |W_c|}{|\mathcal{F}|^\ell} \\
&\geq 1 - |\mathcal{F}|^{-1} - \frac{t \cdot \epsilon_0}{|\mathcal{F}|^k}
\end{aligned}$$

where the last inequality is due to $|B| \geq |\mathcal{F}|^k - |\mathcal{F}|^{k-1}$ and $\sum_{c \in B} |W_c| \leq \epsilon_0 \cdot |\mathcal{F}|^\ell$. Using $\epsilon_0 \leq |\mathcal{F}|^k/10t$, we lower-bound the rejection probability by $0.9 - |\mathcal{F}|^{-1} \geq 0.4$, and the claim follows. ■

Remark 3.10 (on the proof of Proposition 3.9): *We highlight the fact that the core of the proof actually establishes that* if $g = g_{H,V}$ is ϵ-close to linear function that has an image that is a proper subset of $\mathcal{F}^k$, then Step 3 rejects wioth probability at least 0.4.

Remark 3.11 (extension to affine spaces): *In light of Claim 3.2 there is no real need to extend Algorithm 3.6 to the affine case, but let us outline such an extension nevertheless.*

- *The definition of g will be as in the linear case* (i.e., $g(x) = c$ iff $x + cV \in H$)*, except that it will be based on a full-rank k-by-ℓ matrix V such that for some $u \in H$ and every non-zero $c \in \mathcal{F}^k$ it holds that $u + cV \notin H$.* (Indeed, finding such a $u \in H$ is moved from Claim 3.2 to the revised algorithm.)
- *Claim 3.4 is extended to show that H is an $(\ell - k)$-dimensional affine space if and only if g is a affine function with image $\mathcal{F}^k$.*

Recall that testing the affinity of $g : \mathcal{F}^\ell \to \mathcal{F}^k$ can be reduced to testing the linearity of the mapping $x \mapsto g(x) - g(0^\ell)$. Alternatively, one can just use the natural extension of the linearity test of [2] that selects $x, y, z \in \mathcal{F}^\ell$ uniformly at random and checks that $g(x + y) - g(y) = g(x + z) - g(z)$.

3.3 The second reduction

In Section 3.2, for $h : \mathcal{F}^\ell \to \{0,1\}$, we reduced ϵ-testing whether $h^{-1}(1)$ is an $(\ell - k)$-dimensional linear subspace to ϵ-testing the linearity of a function $g : \mathcal{F}^\ell \to \mathcal{F}^k \cup \{\perp\}$, where the value of g at any point can be computed by making $|\mathcal{F}|^k$ queries to h. (Indeed, in order to define the function g, the reduction made $O(|\mathcal{F}|^k)$ additional queries to h.) This yields an ϵ-tester of time complexity $O(|\mathcal{F}|^k/\epsilon)$ for testing $(\ell - k)$-dimensional linear subspaces. In this section we improve this time bound.

Our starting point is the fact that, for every $\epsilon_0 < 1/4$, if g is ϵ_0-close to being a linear function, then it is ϵ_0-close to a unique linear function g', which can be computed by self-correction of g (where each invocation of the self-corrector makes two queries to g and is correct with probability at least $1 - 2\epsilon_0$). Hence, for a constant $\epsilon_0 > 0$ as in Proposition 3.9, if g is ϵ_0-close to a linear function g' with image $\mathcal{F}^k$, then the distance between g and g' equals the distance between h and the corresponding function h' (i.e., $h'(x) = 1$ iff $h'(x) = 0^k$). This suggests a two-step algorithm in which we first invoke Algorithm 3.8 with proximity parameter ϵ_0, and then test equality between h and h', as is detailed next (where we assume again that $\epsilon \leq |\mathcal{F}|^{-k}/O(1)$).

Step I: Invoke Algorithm 3.8 with proximity parameter set to ϵ_0, where $\epsilon_0 > 0$ is a constant as in Proposition 3.9. If the said invocation rejects, then reject. Otherwise, let V be the matrix found in Step 1 of that invocation, and let $g = g_{H,V}$ be the corresponding function.

Let g' denote the *linear function closest to* g, and note that g is ϵ_0-close to g' (or else Algorithm 3.8 would have rejected with high probability). Furthermore, the image of g' equals $\mathcal{F}^k$, or else Algorithm 3.8 would have rejected with probability at least 0.4 (see Remark 3.10). Defining $h' : \mathcal{F}^\ell \to \{0,1\}$ such that $h'(x) = 1$ if and only if $g'(x) = 0^k$, it follows that h' describes an $(\ell - k)$-dimensional linear subspace. Hence, if h is ϵ-far from describing an $(\ell - k)$-dimensional linear subspace, then h is ϵ-far from h'.

Step II: Test whether h equals h' by using a sample of $O(1/\epsilon)$ points. For each sample point, the value of h is obtained by querying h, whereas the value of h' on the sample points is obtained by evaluating g' on these points (since $h'(x) = 1$ iff $g'(x) = 0^k$), where the values of g' on these points are computed via self-correction of g.

The problem is that each query to g is implemented by $|\mathcal{F}|^k$ queries to h. Hence a straightforward implementation will result in making $O(|\mathcal{F}|^k/\epsilon)$ queries to h, which is no better than Algorithm 3.8. Instead, we shall use a sample of $O(1/\epsilon)$ *pairwise-independent* points such that their g'-values are determined by the value of g' at $O(\log(1/\epsilon))$ points, which in turn are computed by self-correction of g that uses $|\mathcal{F}|^k$ queries to h per each point. The details are given in Algorithm 3.12.

Note that if h describes an $(\ell - k)$-dimensional linear subspace, then $g = g'$. On the other hand, if h is ϵ-far from this property and we reached the current step, then h is ϵ-far from h', and a sample of $O(1/\epsilon)$ pairwise-independent points will contain a point of disagreement (w.h.p.).

The key observation here is that Step II can be implemented in complexity $\widetilde{O}(1/\epsilon)$ by taking a sample of $m = O(1/\epsilon)$ *pairwise independent points* in $\mathcal{F}^\ell$ such that evaluating g' on these m points only requires time $O(m + |\mathcal{F}|^k \cdot \widetilde{O}(\log m))$ rather than $O(|\mathcal{F}|^k \cdot m)$ time. This is done as follows.[6]

For $t' = \lceil \log_{|\mathcal{F}|}(m+1) \rceil$, select uniformly $s^{(1)}, ..., s^{(t')} \in \mathcal{F}^\ell$, compute each $g'(s^{(j)})$ via self-correcting g, with error probability $0.01/t'$, and use the sample points $r^{(L)} = L(s^{(1)}, ..., s^{(t')})$ for m non-zero linear function $L : \mathcal{F}^{t'} \to \mathcal{F}$. The key observations are that (1) the $r^{(L)}$'s are pairwise independent, and (2) the values of g' at all $r^{(L)}$'s can be determined based on the values of g' on the $s^{(j)}$'s. This determination is based on the fact that $g'(r^{(L)}) = L(g'(s^{(1)}), ..., g'(s^{(t')})$, by linearity of g'. Hence, the values of g' on t' random points (i.e., the $s^{(j)}$'s) determines the value of g' on $m \leq |\mathcal{F}|^{t'} - 1$ pairwise independent points (i.e., the $r^{(L)}$'s).

Algorithm 3.12 (implementing Step II): *For* $m = O(1/\epsilon)$ *and* $t' = \lceil \log_{|\mathcal{F}|}(m+1) \rceil$, *set* $t'' = O(1) + \log_2 t'$, *and proceed as follows.*

[6] Inspired by [5] (as presented in [3, Sec. 7.1.3] for $\mathcal{F} = \mathrm{GF}(2)$). The salient feature of this sample space is that the values of any linear function at all points in the sample (i.e., the $r^{(L)}$'s) can be determined by the values of this function at very few points (i.e., the $s^{(j)}$'s).

1. *Select uniformly $s^{(1)}, ..., s^{(t')} \in \mathcal{F}^\ell$.*

2. *For each $j \in [t']$, select uniformly $w^{(1)}, ..., w^{(t'')} \in \mathcal{F}^\ell$, and set $\sigma^{(j)}$ to equal the majority vote of $g(s^{(j)} + w^{(1)}) - g(w^{(1)}), ..., g(s^{(j)} + w^{(t'')}) - g(w^{(t'')})$, where the values of g at each point x is determined according to the value of h at the points $\{x + cV : c \in \mathcal{F}^k\}$.*

   Recall that $g(x) = c$ is c is the unique point in $\mathcal{F}^k$ such that $h(x + cV) = 1$, and is set to $\perp$ otherwise. If the value of g at any point is set to $\perp$, we can abort this algorithm and reject h. Alternatively, we can set $\sigma^{(j)}$ to $\perp$, and define the $z + \perp = \perp$ for every $z \in \mathcal{F}^k$.

   (Indeed, $\sigma^{(j)}$ is our sound guess for $g'(s^{(j)})$, and this guess is correct with probability $1 - \exp(-\Omega(t'')) > 1 - 0.01/t'$).

3. *For each of m non-zero linear function $L : \mathcal{F}^{t'} \to \mathcal{F}$, let $r^{(L)} = L(s^{(1)}, ..., s^{(t')})$ and check whether $h(r^{(L)})$ equals our guess for $h'(r^{(L)})$, where the later value is set to 1 if and only if $L(\sigma^{(1)}, ..., \sigma^{(t')}) = 0^k$. Accept if and only if all checks were successful* (i.e., equality holds in all).

   (Recall that $g'(r^{(L)}) = g'(L(s^{(1)}, ..., s^{(t')})) = L(g'(s^{(1)}), ..., g'(s^{(t')}))$. Hence, $L(\sigma^{(1)}, ..., \sigma^{(t')})$ is our sound guess for $g'(r^{(L)})$, and this guess is correct if all guesses for the $g'(s^{(j)})$'s are correct, which happens with probability 0.99).

The time complexity of Algorithm 3.12 is $O(t' \cdot t'' \cdot |\mathcal{F}|^k + m) = \widetilde{O}(\log_{|\mathcal{F}|}(1/\epsilon)) \cdot |\mathcal{F}|^k + O(1/\epsilon)$. This dominates the time complexity of Step I, which is $O(|\mathcal{F}|^k/\epsilon_0) = O(|\mathcal{F}|^k)$. Recall that for $\epsilon > 4 \cdot |\mathcal{F}|^{-k}$ there is an almost trivial tester of complexity $O(1/\epsilon)$, and note that for $\epsilon < |\mathcal{F}|^{-k-1.01 \cdot \log_{|\mathcal{F}|} k}$ it holds that $\widetilde{O}(\log_{|\mathcal{F}|}(1/\epsilon)) \cdot |\mathcal{F}|^k = O(1/\epsilon)$. Hence, our complexity bound is (slightly) inferior to the optimal bound of $O(1/\epsilon)$ only for a narrow range of parameters (i.e., for $\epsilon \in [k^{-1.01} \cdot |\mathcal{F}|^{-k}, 4 \cdot |\mathcal{F}|^{-k}]$).

Theorem 3.13 (analysis of the foregoing algorithm): *Consider an algorithm that invokes the algorithm captured by the foregoing Steps I and II, where Step II is as detailed in Algorithm 3.12, for a constant number of times and accept if and only if at least two third of the invocations accepted. Then, the resulting algorithm constitutes a tester for $(\ell - k)$-dimensional linear subspaces.*

Note that this tester has two-sided error probability. Obtaining an analogous one-sided error tester is left as an open problem. It is indeed possible that such a tester does not exist.

Proof: We consider a single invocation of Steps I and II. If h describes an $(\ell - k)$-dimensional linear subspace, then (w.h.p.) the execution reaches Step II, which always accepts. On the other hand, if h is ϵ-far from describing an $(\ell - k)$-dimensional linear subspace, then we consider two cases.

1. If h is ϵ_0-far from describing an $(\ell-k)$-dimensional linear subspace, then Step I rejects (w.h.p).

2. Otherwise, assuming that Step II is reached, we consider the corresponding functions g and g'. Recall that the image of g' equals $\mathcal{F}^k$, since otherwise Step I rejects with probability at least 0.4 (see Remark 3.10). Hence, h must be ϵ-far from h', since in this case h' describes an $(\ell - k)$-dimensional linear subspace.

   In this case (assuming Step II is reached), with probability at least 0.99, the tester obtains the correct values of g' at all $s^{(j)}$'s and hence determined correctly the values of g' at all the $r^{(L)}$'s. Since these $r^{(L)}$ are uniformly distributed in $\mathcal{F}^\ell$ in a pairwise independent manner, with probability at least $1 - \frac{m\epsilon}{(m\epsilon)^2} > 0.9$, the sample contains a point on which h and h' disagree.

In conclusion, if h is ϵ-far from the tested property, then the foregoing algorithm rejects with probability at least 0.4. Using the threshold decision rule (i.e., accepting if at least two thirds of the invocations of the foregoing algorithm accept), the theorem follows. ■

Remark 3.14 (extension to affine spaces): *Again, there is no real need to extend the foregoing to the affine case, but we outline such an extension nevertheless. We first note that self-correction of an affine g requires querying it at three random locations rather than at two; specifically, to obtain $g'(s)$, we select uniformly $r, r' \in \mathcal{F}^\ell$ and query g at $s+r+r', r, r'$, while relying on $g'(s) = g'(s+r-r') - g'(r) + g'(r')$. Likewise, the equality $g'(r^{(L)}) = L(g'(s^{(1)}), ..., g'(s^{(t')})$ is replaced by $g'(r^{(L)}) = L(g'(s^{(1)}), ..., g'(s^{(t')}) - L(v, ..., v) + v$, where $v = g'(0^\ell)$ is also obtained by self-correction of g, which calls for making $3 \cdot |\mathcal{F}|^k$ additional queries.*

4 On testing monotone monomials

As stated in the introduction, the problem that motivated our study is trying to reduce testing monomials to testing dictatorships. Such a reduction is preferable to an extension of the ideas that underly the tests of (monotone) dictatorship towards testing the set of functions that are (monotone) k-monomials, for any $k \geq 1$. In this section, we first review the said extension, as performed in Parnas, Ron, and Samorodnitsky [6], and then we present our alternative. We start with the definition of the relevant properties.

Definition 4.1 (monomial and monotone monomial): *A Boolean function $f : \{0,1\}^\ell \to \{0,1\}$ is called a* k-monomial *if for some k-subset $I \subseteq [\ell]$ and $\sigma = \sigma_1 \cdots \sigma_\ell \in \{0,1\}^\ell$ it holds $f(x) = \wedge_{i \in I}(x_i \oplus \sigma_i)$. It is called a* monotone k-monomial *if $\sigma = 0^\ell$.*

Indeed, the definitions of (regular and monotone) 1-monomials coincide with the notions of (regular and monotone) dictatorships. We focus on the task of testing monotone k-monomials, while recalling that the task of testing k-monomials is reducible to it (see [6] or [4, Sec. 5.2.2.1]). We also recall that this testing problem is of interest only when the proximity parameter, denoted ϵ, is small (in relation to 2^{-k}). In contrast, when $\epsilon > 2^{-k+2}$, we may just estimate the density of $f^{-1}(1)$ and accept if and only if the estimate is below $\epsilon/2$.

4.1 The tester of Parnas, Ron, and Samorodnitsky

We start by interpreting the dictatorship tester of [1, 6] in a way that facilitates its generalization. Recall that these works perform a dictatorship test by first testing that the function is linear and then performing a "conjunction check" (i.e., checking that $f(x \wedge y) = f(x) \wedge f(y)$). Now, if f is a monotone dictatorship, then $f^{-1}(1)$ is an $(\ell-1)$-dimensional affine subspace (of the ℓ-dimensional space $\{0,1\}^\ell$), where $\{0,1\}$ is associated with the two-element field GF(2). Specifically, if $f(x) = x_i$, then this subspace is $\{x \in \{0,1\}^\ell : x_i = 1\}$. In this case, the *linearity tester* could be thought of as testing that $f^{-1}(1)$ is an arbitrary $(\ell-1)$-dimensional affine subspace, whereas the "conjunction check" verifies that this subspace is an affine translation by 1^ℓ of a linear space that is spanned by $\ell - 1$ unit vectors (i.e., vectors of Hamming weight 1).[7]

[7] That is, we requires that this subspace has the form $\left\{1^\ell + \sum_{j \in ([\ell] \setminus \{i\})} c_j e_j : c_1, ..., c_\ell \in \{0,1\}\right\}$, where $e_1, ..., e_\ell \in \{0,1\}^\ell$ are the ℓ unit vectors (i.e., vectors of Hamming weight 1).

When generalizing the treatment for abitrary k, we observe that if f is a monotone k-monomial, then $f^{-1}(1)$ is an $(\ell-k)$-dimensional affine subspace. So the foregoing two-step procedure generalizes to first testing that $f^{-1}(1)$ is an $(\ell-k)$-dimensional affine subspace, and then testing that it is an affine subspace of the right form (i.e., it has the form $\{x\in\{0,1\}^\ell : (\forall i\in I)\ x_i=1\}$, for some k-subset I). Following are outlines of the treatment of these two tasks in [6].

Testing affine subspaces. Supposed that the alleged affine subspace H is presented by a Boolean function h such that $h(x)=1$ if and only if $x\in H$. (Indeed, in our application, $h=f$.) We wish to test that H is indeed an affine subspace.

(Actually, we are interested in testing that H has a given dimension, but this extra condition can be checked easily by estimating the density of H in $\{0,1\}^\ell$, since we are willing to have complexity that is inversely proportional to the designated density (i.e., 2^{-k}).)[8]

This task is related to linearity testing and it was indeed solved in [6] using a tester and an analysis that resembles the standard linearity tester of [2]. Specifically, the tester selects uniformly $x,y\in H$ and $z\in\{0,1\}^\ell$ and checks that $h(x+y+z)=h(z)$ (i.e., that $x+y+z\in H$ if and only if $z\in H$). Indeed, we uniformly sample H by repeatedly sampling $\{0,1\}^\ell$ and checking whether the sampled element is in H.

Note that, for co-dimension $k>1$, the function h is not affine (i.e., $h(x)=h(y)=0$, which means $x,y\notin H$, does not determine the value of $h(x+y)$ (i.e., whether $x+y\in H$)).[9] Still, testing affine subspaces can be reduced to testing linearity (albeit not of h but rather of a related function), providing an alternative to the presentation of [6]. Presenting such a reduction is the core of this paper (see Section 3, which handles an arbitrary finite field $\mathcal{F}$ whereas the current application only requires $\mathcal{F}=\mathrm{GF}(2)$).

Testing that an affine subspace is a translation by 1^ℓ of a linear subspace spanned by unit vectors. Suppose that an affine subspace H' is presented by a Boolean function, denoted h', and that we wish to test that H' has the form $\left\{1^\ell+\sum_{i\in[\ell]\setminus I}c_ie_i : c_1,...,c_\ell\in\{0,1\}\right\}$, where $e_1,...,e_\ell\in\{0,1\}^\ell$ are unit vectors, and $I\subseteq[\ell]$ is arbitrary. That is, we wish to test that $h'(x)=\wedge_{i\in I}x_i$.

This can be done by picking uniformly $x\in H'$ and $y\in\{0,1\}^\ell$, and checking that $h'(x\wedge y)=h'(y)$ (i.e., $x\wedge y\in H'$ if and only if $y\in H'$). Note that if H' has the form $1^\ell+L$, where L is a linear subspace spanned by the unit vectors $\{e_i : i\in[\ell]\setminus I\}$ for some I, then $h'(z)=\wedge_{i\in I}z_i$ holds for all $z\in\{0,1\}^\ell$, and $h'(x\wedge y)=h'(x)\wedge h'(y)$ holds for all $x,y\in\{0,1\}^\ell$. On the other hand, as shown in [6], if H' is an affine subspace that does not have the foregoing form, then the test fails with probability at least 2^{-k-1}.

However, as in the case of $k=1$, we do not have access to h' but rather to a Boolean function h that is (very) close to h'. So we need to obtain the value of h' at specific points by querying h at uniformly distributed points. Specifically, the value of h' at z is obtained by uniformly selecting $r,s\in h^{-1}(1)$ and using the value $h(z+r-s)$. In other words, we self-correct h at any desired point z by using the value of h at a point obtained by shifting z by the difference between two random elements of $h^{-1}(1)$, while hoping that these points actually reside in the affine subspace H' (so that their difference is in the linear space $H'-H'$). This hope is likely to materialize when h is $0.01\cdot 2^{-k}$-close to h'.

[8] Recall that if $\epsilon<2^{-k+2}$, then $O(2^k)=O(1/\epsilon)$, and otherwise (i.e., for $\epsilon\geq 2^{-k+2}$) testing affinity of H reduces to estimating the density of $h^{-1}(1)$.

[9] In contast, if $h(x)=1$, then $h(x+y)=h(y)$, which means that $x+y\in H$ iff $y\in H$ (when $x\in H$).

The foregoing is indeed related to the conjunction check performed as part of the dictatorship tester of [1, 6], and the test and the analysis in [6] (for the case of $k > 1$) resemble the corresponding parts in [1, 6] (which handle the case of $k = 1$).

In contrast, building on the main idea of Section 3, in Section 4.2, we present a simple reduction from the general case (of any $k \geq 1$) to the special case (of $k = 1$).

4.2 An alternative tester of monotone monomials

As outlined in Section 4.1, the function $f : \{0,1\}^\ell \to \{0,1\}$ is a monotone k-monomial if and only if f describes an $(\ell - k)$-dimensional affine space that is a translation by 1^ℓ of an $(\ell - k)$-dimensional axis-parallel linear space; that is, if $f^{-1}(1)$ has the form $\{yG + 1^\ell : y \in \{0,1\}^{\ell-k}\}$, where G is a full-rank $(\ell - k)$-by-ℓ Boolean matrix that contains k all-zero columns. Hence, we may focus on testing that the function $h : \{0,1\}^\ell \to \{0,1\}$ defined by $h(x) \stackrel{\rm def}{=} f(x + 1^\ell)$ describes an $(\ell - k)$-dimensional *axis-parallel* linear space. (Indeed, the reduction of Section 3.1 is instantiated here by mandating $u = 1^\ell$.)

Following [6], we first test that the Boolean function h describes an $(\ell - k)$-dimensional linear space, and next test that this linear space has the right form. Again, we assume that the proximity parameter ϵ is upper-bounded by $\epsilon_0 \cdot 2^{-k}$, for some constant $\epsilon_0 > 0$. Hence, if h passes the first test, then we may assume that h is ϵ-close to Boolean function h' that describes an $(\ell - k)$-dimensional linear space. Defining corresponding functions $g : \{0,1\}^\ell \to \{0,1\}^k$ and $g' : \{0,1\}^\ell \to \{0,1\}^k$ as in Section 3.2, we infer that g is $2^k \cdot \epsilon$-close to the linear function g', which has image $\{0,1\}^k$, while noting that $2^k \cdot \epsilon \leq \epsilon_0$. (Indeed, we may use the function $g = g_{h^{-1}(1),V}$ defined in Step 1 of Algorithm 3.6, or just run this step anew.)

The key observation is that h' describes an axis-parallel linear space (i.e., the set $\{x \in \{0,1\}^\ell : h'(x) = 1\}$ equals the linear space $\{yG : y \in \{0,1\}^{\ell-k}\}$ for a full-rank matrix G with k all-zero columns) if and only if g' is a projection function (i.e., $g'(x) = x_I$ for some k-subset I).

At this point, we generalize the observation that underlies the conjunction check that is part of the dictatorship test of [1, 6]. Specifically, given that g' is a linear function with image $\{0,1\}^k$, we test that $g'(x) = x_I$ (for some k-subset I) by selecting uniformly $r, s \in \{0,1\}^\ell$ and checking whether $g'(r)g'(s) = g'(rs)$, where uv denotes the bit-by-bit produce of u and v. The point is that the analysis of this test can be reduced to the analysis of the conjunction check by considering the k bits in the output of g'. Details follow.

Let $g'_i(x)$ denote the $i^{\rm th}$ bit of $g'(x)$. On the one hand, if $g'(x) = x_I$ for some k-subset I, then for each $i \in [k]$ the function g'_i is a dictatorship (i.e., $g'_i(x) = x_{j_i}$ for some $j_i \in [\ell]$), and so $g'_i(rs) = (rs)_{j_i} = g'_i(r)g'_i(s)$ for all $r, s \in \{0,1\}^\ell$. Hence, in this case, $g'(rs) = g'(r)g'(s)$ holds for all $r, s \in \{0,1\}^\ell$. On the one hand, if g' is not a projection function, then (using the fact that g' is linear and its image equals $\{0,1\}^k$) there exists $i \in [k]$ such that the function g'_i is a linear combination of at least two bits. The known analysis (cf. [1, 6]) implies that in this case $\mathbf{Pr}_{r,s}[g'_i(rs) = g'_i(r)g'_i(s)] \leq 3/4$, which implies $\mathbf{Pr}_{r,s}[g'(rs) = g'(r)g'(s)] \leq 3/4$.

However, as in Section 3, we do not have access to g', but rather obtain its values at desired points by applying self-correction to g, which is ϵ_0-close to g'. We can afford to compute g at any desired point, since we intend to do so only a constant number of times. Specifically, it suffices to perform the foregoing "conjunction" check once, since such an execution rejects an improper h (i.e., one that is close to a function h' that describes a linear space that is not axis-parallel) with probability at least $0.25 - 4\epsilon_0 > 0.2$ (assuming $\epsilon_0 > 0$ is sufficiently small). To wrap-up, we obtain the following tester (where we assume again that $\epsilon \leq \epsilon_0/2^k$, for some adequate constant $\epsilon_0 > 0$).

Algorithm 4.2 (testing whether f is a monotone k-monomial): *On input a proximity parameter*

$\epsilon \in (0, \epsilon_0 \cdot 2^{-k}]$ *and oracle access to* $f : \{0,1\}^\ell \to \{0,1\}$, *the algorithm proceeds as follows.*

1. *Apply the tester of Section 3.3 to test whether* $h : \{0,1\}^\ell \to \{0,1\}$, *defined by* $h(x) \stackrel{\rm def}{=} f(x+1^\ell)$, *describes an* $(\ell - k)$*-dimensional linear space over* GF(2). *The said tester is invoked with proximity parameter* ϵ, *and each query* x *is answered by the value* $f(x+1^\ell)$. *If the foregoing tester rejects, then the current algorithm reject.*
2. *Find a matrix* V *as in Step 1 of Algorithm 3.6, and let* $g = g_{h^{-1}(1),V} : \{0,1\}^\ell \to \{0,1\}^k$ *denote the corresponding function. Select uniformly* $r, s, w \in \{0,1\}^\ell$ *and accept if and only if* $g(r)g(s) = g(rs+w) - g(w)$, *where* $g(rs+w) - g(w)$ *represents self-correcting the value of* g *at* rs. (Recall that uv denotes the bit-by-bit produce of u and v, and that the value of g at x is computed by querying h at the points $\{x + cV : c \in \{0,1\}^k\}$.)

The complexity of Algorithm 4.2 is dominated by the complexity of Step 1, which is $\widetilde{O}(\log(1/\epsilon)) \cdot 2^k + O(1/\epsilon)$.

Proposition 4.3 (analysis of Algorithm 4.2): *Assuming that* $\epsilon \leq \epsilon_0 \cdot 2^{-k}$, *the following holds.*

1. *Algorithm 4.2 accepts any monotone* k*-monomial with probability at least* 2/3.
2. *Algorithm 4.2 rejects any function that is* ϵ*-far from being a monotone* k*-monomial with probability at least* 0.2.

The error probability in the case that f is a monotone k-monomial is due to Step 1. (Indeed, in this case $g(x) = x_I$ for some k-subset I, and so Step 2 accepts with probability 1.) The analysis of the case of functions that are ϵ-far from being monotone k-monomials reduces to the analysis of Step 2, in which we may assume that h is ϵ-close to describing an $(\ell - k)$-dimensional linear space. In this case, g is ϵ-close to a linear function g' that is not a projection function and has image $\{0,1\}^k$. Using the fact that g is ϵ_0-close to g', it follows that, with probability at least $1 - 4\epsilon_0$ (over the choice of r, s and w), it holds that $g(r) = g'(r)$, $g(s) = g'(s)$, $g(rs+w) = g'(rs+w)$, and $g(w) = g'(w)$. Using $g'(rs+w) - g'(w) = g'(rs)$, we have

$$\begin{aligned}\mathbf{Pr}_{r,s,w}[g(r)g(s) \neq g(rs+w) - g(w)] &\geq \mathbf{Pr}_{r,s,w}[g'(r)g'(s) \neq g'(rs+w) - g'(w)] - 4\epsilon_0 \\ &\geq \mathbf{Pr}_{r,s,w}[g'_i(r)g'_i(s) \neq g'_i(rs)] - 4\epsilon_0,\end{aligned}$$

for any $i \in [k]$ (where $g'_i(x)$ denotes the $i^{\rm th}$ bit of $g'(x)$). Using the hypothesis g' is a linear function with image $\{0,1\}^k$ that is not projection function, we can pick i such that $g'_i : \{0,1\}^\ell \to \{0,1\}$ is neither a dictatorship nor a constant function. In this case $g'_i(x) = \sum_{j \in J} x_j$ for some subset J of size at least two, and $\mathbf{Pr}_{r,s,w}[g'_i(r)g'_i(s) \neq g'_i(rs)] \geq 1/4$ follows. This completes the proof of Proposition 4.3.

Comparison to [6]. Algorithm 4.2 differs from the tester of [6] in two aspects. Firstly, Step 1 uses the tester of Section 3.3, which is based on a reduction of testing affine spaces to testing linear functions. Specifically, we reduce testing that $f : \{0,1\}^\ell \to \{0,1\}$ describes an affine space to testing that a related $h : \{0,1\}^\ell \to \{0,1\}$ describes a linear space, which in turn is reduced to testing that a related $g : \{0,1\}^\ell \to \{0,1\}^k$ is linear. In contrast, testing affine spaces is performed in [6] by modifying the linearity tester of [2] and mimicking the known analysis of this tester.

Second, Step 2 uses the foregoing function g (of Step 1), and reduces testing that the linear space h has the right form to testing that g satisfies a conjunction condition that generalizes the condition used in the case of $k = 1$. Furthermore, the analysis of this test is reduced to the analysis of the case $k = 1$. Again, as can be seen in Section 4.1, the path taken by [6] involves a modification of the procedure and analysis used in the case of $k = 1$.

Acknowledgements

I am grateful to Roei Tell for reading prior versions of this text and pointing out numerous inaccuracies and gaps. I also wish to thank Clement Canonne and Tom Gur for their comments. This research was partially supported by the Israel Science Foundation (grant No. 671/13).

References

[1] M. Bellare, O. Goldreich and M. Sudan. Free Bits, PCPs and Non-Approximability – Towards Tight Results. *SIAM Journal on Computing*, Vol. 27, No. 3, pages 804–915, 1998. Extended abstract in *36th FOCS*, 1995.

[2] M. Blum, M. Luby and R. Rubinfeld. Self-Testing/Correcting with Applications to Numerical Problems. *Journal of Computer and System Science*, Vol. 47, No. 3, pages 549–595, 1993. Extended abstract in *22nd STOC*, 1990.

[3] O. Goldreich. *Computational Complexity: A Conceptual Perspective.* Cambridge University Press, 2008.

[4] O. Goldreich. *Introduction to Property Testing.* Cambridge University Press, 2017.

[5] O. Goldreich and L.A. Levin. A hard-core predicate for all one-way functions. In the proceedings of *21st ACM Symposium on the Theory of Computing*, pages 25–32, 1989.

[6] M. Parnas, D. Ron, and A. Samorodnitsky. Testing Basic Boolean Formulae. *SIAM Journal on Disc. Math. and Alg.*, Vol. 16 (1), pages 20–46, 2002.