Linearity Testing/Testing Hadamard Codes (1990; Blum, Luby Rubinfeld)

Ronitt Rubinfeld, MIT, theory.lcs.mit.edu/˜ronitt

entry editor: Venkatesan Guruswami

INDEX TERMS: Property testing.
SYNONYMS: Linearity testing, testing Hadamard codes, homomorphism testing.

1 PROBLEM DEFINITION

This problem is concerned with distinguishing functions that are homomorphisms, i.e. satisfying $\forall x, y,\ f(x) + f(y) = f(x + y)$, from those functions that must be changed on at least ϵ fraction of the domain in order to be turned into a homomorphism, given query access to the function. This problem was initially motivated by applications to testing programs which compute linear functions [8]. Since Hadamard codes are such that the codewords are exactly the evaluations of linear functions over boolean variables, a solution to this problem gives a way of distinguishing codewords of the Hadamard code from those strings that are far in relative Hamming distance from codewords. These algorithms were in turn used in the constructions of Probabilistically Checkable Proof Systems (cf. [3]). Further work has extended these techniques to testing other properties of low degree polynomials and solutions to other addition theorems [3, 9, 24, 25].

Notations For two finite groups G, H (not necessarily Abelian), an arbitrary map $f : G \to H$ is a *homomorphism* if

$$\forall x, y,\ f(x) \times f(y) = f(x \times y).$$

f is *ϵ-close to a homomorphism* if there is some homomorphism g such that g and f differ on at most $\epsilon|G|$ elements of G, and f is *ϵ-far* otherwise.

Given a parameter $0 \le \epsilon \le 1$, and query access to a function $f : G \to H$, a *homorphism tester* is an algorithm $\mathcal{T}$ which outputs "Pass" if f is a homomorphism, and "Fail" if f is ϵ-far from a homomorphism. The homomorphism tester should err with probability at most 1/3 for any f.[1]

For two finite groups G, H (not necessarily Abelian), an arbitrary map $f : G \to H$, and a parameter $0 < \epsilon < 1$, define δ_f (the subscript is dropped and this is referred to as δ, when f is obvious from the context) then the *probability of group law failure*, by

$$1 - \delta = \Pr_{x,y}\left[f(x) \times f(y) = f(x \times y)\right].$$

Define τ such that τ is the minimum ϵ for which f is ϵ-close to a homomorphism.

Problem 1. *For f, δ as above, is it possible to upper bound τ in terms of a function that depends on the probability of group law failure δ, but not on the size of the domain $|G|$?*

[1] The choice of 1/3 is arbitrary. Using standard techniques, any homomorphism tester satisfying 1/3 error probability can be turned into a homomorphism tester with $0 < \beta < 1/3$ error probability by repeating the original tester $O(\log \frac{1}{\beta})$ times and taking the majority answer.

2 KEY RESULTS

Blum, Luby and Rubinfeld [8], considered this question and showed that over cyclic groups, there is a constant δ_0, such that if $\delta \leq \delta_0$, then the one can upper bound ϵ in terms of a function of δ that is independent of $|G|$. This yields a homomorphism tester with query complexity that depends (polynomially) on $1/\epsilon$, but is independent of $|G|$. The final version of [8] contains an improved argument due to Coppersmith [10], which applies to all Abelian groups, shows that $\delta_0 < 2/9$ suffices, and that ϵ is upper bounded by the smaller root of $x(1-x) = \delta$ (yielding a homomorphism tester with query complexity linear in $1/\epsilon$). Furthermore, the bound on δ_0 was shown to be tight for general groups [10].

In [6], a relationship between the probability of group law failure and the closeness to being a homomorphism was established that applies to general (non-Abelian) groups. For a given δ, let $\tau = (3 - \sqrt{9-24\delta})/12 \leq \delta/2$ be the smaller root of $3x - 6x^2 = \delta$. In [6] it is shown that for $\delta_0 < 2/9$, then f is τ-close to a homomorphism. The condition on δ, and the bound on τ as a function of δ, are shown to be tight. The latter improves on the relationship given in [8, 10].

There has been interest in improving various parameters of homomorphism testing results, due to their use in the construction of Probabilistically Checkable Proof Systems (cf. [3]). In particular, both the constant δ_0 and the number of random bits required by the homomorphism test affect the efficiency of the proof system and in turn the hardness of approximation results that one can achieve using the proof system.

The homomorphism testing results can be improved in some cases: It has been previously mentioned that $\delta_0 < 2/9$ is optimal over general Abelian groups [10]. However, using Fourier techniques, Bellare et. al. [5] have shown that for groups of the form $(\mathbf{Z}/2)^n$, $\delta_0 \leq 45/128$ suffices. For such δ_0, $\epsilon < \delta$. Kiwi later provided a similar result based on the discrete Fourier transform and weight distributions to improve the bound on the dependence of ϵ on δ [17].

Several works have shown methods of reducing the number of random bits required by the homomorphism tests. That is, in the natural implementation of the homomorphism test, $2\log|G|$ random bits per trial are used to pick x, y. The results of [7, 14, 26, 28, 29] have shown that fewer random bits are sufficient for implementing the homomorphism tests. In particular, Trevisan [29] and Samorodnitsky and Trevisan [26] have considered the "amortized query complexity" of testing homomorphisms, which is a measure that quantifies the trade-off between the query complexity of the testing algorithm and the probability of accepting the function. Linearity tests with low amortized query complexity are useful in constructing PCP systems with low amortized query complexity. A simpler analysis which improves the dependence of the acceptance probability in terms of the distance of the tested function to the closest linear function is given in [14]. The work of [28] gives a homomorphism test for general (non-Abelian) groups that uses only $(1+o(1))\log_2|G|$ random bits. Given a Cayley graph that is an expander with normalized second eigenvalue γ, and for the analogous definitions of δ, τ, they show that for $\delta < (1-\gamma)/12$, τ is upper bounded by $4\delta/(1-\gamma)$. Very recently, Samordnitsky and Trevisan [27] have considered a relaxed version of a homomorphism test which accepts linear functions and rejects functions with low influences.

The case when G is a subset of an infinite group, f is a real-valued function and the oracle query to f returns a finite precision approximation to $f(x)$ has been considered in [2, 11, 12, 20, 21], and testers with query complexity that are independent of the domain size have been given (see [19] for a survey).

A related problem on convolutions of distributions In the following, a seemingly unrelated question about distributions that are close to their self-convolutions is mentioned: Let $A = \{a_g | g \in G\}$ be a distribution on group G. The convolution of distributions A, B is

$$C = A * B,\ c_x = \sum_{y,z \in G;\ yz=x} a_y b_z.$$

Let A' be the *self-convolution* of A, $A * A$, i.e. $a'_x = \sum_{y,z \in G; yz=x} a_y a_z$. It is known that $A = A'$ exactly when A is the uniform distribution over a subgroup of G. Suppose it is known that A is close to A', can one say anything about A in this case? Suppose $dist(A, A') = \frac{1}{2} \sum_{x \in G} |a_x - a'_x| \leq \epsilon$ for small enough ϵ. Then [6] show that A must be close to the uniform distribution over a subgroup of G. More precisely, in [6] it is shown that for a distribution A over a group G, if $dist(A, A') = \frac{1}{2} \sum_{x \in G} |a_x - a'_x| \leq \epsilon \leq 0.0273$, then there is a subgroup H of G such that $dist(A, U_H) \leq 5\epsilon$, where U_H is the uniform distribution over H [6]. On the other hand, in [6] there is an example of a distribution A such that $dist(A, A * A) \approx .1504$, but A is not close to uniform on any subgroup of the domain.

A weaker version of this result, was used to prove a preliminary version of the homomorphism testing result in [8]. To give a hint of why one might consider the question on convolutions of distributions when investigating homomorphism testing, consider the distribution A_f achieved by picking x uniformly from G and outputting $f(x)$. It is easy to see that the error probability δ in the homomorphism test is at least $dist(A_f, A_f * A_f)$. The other, more useful, direction is less obvious. In [6] it is shown that this question on distributions is "equivalent" in difficulty to homomorphism testing:

Theorem 1. *Let G, H be finite groups. Assume that there is a parameter β_0 and function ϕ such that the following holds:*

> *For all distributions A over group G, if $dist(A * A, A) \leq \beta \leq \beta_0$ then A is $\phi(\beta)$-close to uniform over a subgroup of G.*

Then, for any $f : G \to H$ and $\delta < \beta_0$ such that $1 - \delta = Pr[f(x) * f(y) = f(x * y)]$, and $\phi(\delta) \leq 1/2$, it is the case that f is $\phi(\delta)$-close to a homomorphism.

3 APPLICATIONS

Self-testing/correcting programs When a program has not been verified and therefore is not known to be correct on all inputs (or possibly even known to be incorrect on some inputs), [8] have suggested the following approach: take programs that are known to be correct on most inputs and apply a simple transformation to produce a program that is correct on every input. This transformation is referred to as producing a *self-corrector*. Moreover, for many functions, one can actually test that the program for f is correct on most inputs, without the aid of another program for f that has already been verified. Such testers for programs are referred to as *self-testers*.

The linearity testing problem was initially motivated by applications to constructing self-testers for programs which purport to compute various linear functions [8]. Such functions include integer, polynomial, matrix and modular multiplication and division. Once it is verified that a program agrees on most inputs with a specific linear function, the task of determining whether it agrees with the *correct* linear function on most inputs becomes much easier.

Furthermore, for programs purporting to compute linear functions, it is very simple to construct self-correctors: Assume one is given a program which on input x outputs $f(x)$, such that f agrees on most inputs with linear function g. Consider the algorithm that picks $c \log 1/\beta$ values y, computes $f(x + y) - f(y)$ and outputs the value that is seen most often. If f is $\frac{1}{8}$-close to g, then since both y and $x + y$ are uniformly distributed, it is the case that for at least $3/4$ of the choices of y, $g(x + y) = f(x + y)$ *and* $g(y) = f(y)$, in which case $f(x + y) - f(y) = g(x)$. Thus it is easy to show that there is a constant c such that if f is $\frac{1}{8}$-close to a homomorphism g, then for all x, the above algorithm will output $g(x)$ with probability at least $1 - \beta$.

Probabilistically Checkable Proofs An equivalent formulation of the linearity testing problem is in terms of the query complexity of testing a codeword of a Hadamard code. The results mentioned about have been used to construct Probabilistically Checkable Proof systems which can be verified with very few queries (cf. [3, 13]).

4 OPEN PROBLEMS

It is natural to wonder what other classes of functions have testers whose efficiency is sublinear in the domain size? There are some partial answers to this question: The field of functional equations is concerned with the prototypical problem of characterizing the set of functions that satisfy a given set of properties (or functional equations). For example, the class of functions of the form $f(x) = \tan Ax$ are characterized by the functional equation

$$\forall x, y,\ f(x+y) = \frac{f(x)+f(y)}{1-f(x)f(y)}.$$

D'Alembert's equation

$$\forall x, y,\ f(x+y) + f(x-y) = 2f(x)f(y)$$

characterizes the functions $0, \cos Ax, \cosh Ax$. Multivariate polynomials of total degree d over $\mathcal{Z}_p$ for $p > md$ can be characterized by the equation

$$\forall \hat{x}, \hat{h} \in Z_p^m, \sum_{i=0}^{d+1} \alpha_i f(\hat{x} + i\hat{h}) = 0,$$

where $\alpha_i = (-1)^{i+1}\binom{d+1}{i}$. All of the above characterizations are known to yield testers for the corresponding function families whose query complexity is independent of the domain size (though for the case of polynomials, there is a polynomial dependence on the total degree d) [9, 24, 25]. A long series of works have given increasingly efficient to test characterizations of functions that are low total degree polynomials (cf. [1, 3, 4, 15, 18, 22, 23]).

A second line of questions that remain to be understood regard which codes are such that strings can be quickly tested to determine whether they are close to a codeword? Some initial work on this questions is given in [1, 15, 16, 18].

5 EXPERIMENTAL RESULTS

None is reported.

6 DATA SETS

None is reported.

7 URL to CODE

None is reported.

8 CROSS REFERENCES

Decoding Hadamard Codes and Learning Heavy Fourier Coefficients of Boolean Functions. Learning Heavy Fourier Coefficients over Z_N.

References

[1] N. Alon, T. Kaufman, M. Krivilevich, S. Litsyn, and D. Ron, *Testing low-degree polynomials over gf(2)*, in Proceedings of RANDOM '03, 2003, pp. 188–199.

[2] S. Ar, M. Blum, B. Codenotti, and P. Gemmell, *Checking approximate computations over the reals*, in Proceedings of the Twenty-Fifth Annual ACM Symposium on the Theory of Computing, 2003, pp. 786–795.

[3] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy, *Proof verification and the hardness of approximation problems*, J. ACM, 45 (1998), pp. 501–555.

[4] S. Arora and M. Sudan, *Improved low degree testing and its applications*, in Proceedings of the Twenty-Ninth Annual ACM Symposium on the Theory of Computing, 1997, pp. 485–495.

[5] M. Bellare, D. Coppersmith, J. Håstad, M. Kiwi, and M. Sudan, *Linearity testing over characteristic two*, IEEE Transactions on Information Theory, 42 (1996), pp. 1781–1795.

[6] M. Ben-Or, D. Coppersmith, M. Luby, and R. Rubinfeld, *Non-abelian homomorphism testing, and distributions close to their self-convolutions.*, in Proceedings of APPROX-RANDOM, 2004, pp. 273–285.

[7] E. Ben-Sasson, M. Sudan, S. Vadhan, and A. Wigderson, *Randomness-efficient low degree tests and short pcps via epsilon-biased sets*, in Proceedings of the Thirty-Fifth Annual ACM Symposium on the Theory of Computing, 2003, pp. 612–621.

[8] M. Blum, M. Luby, and R. Rubinfeld, *Self-testing/correcting with applications to numerical problems*, JCSS, 47 (1993), pp. 549–595.

[9] R. Cleve and M. Luby, *A note on self-testing/correcting methods for trigonometric functions*, in International Computer Science Institute Technical Report TR-90-032, 1990.

[10] D. Coppersmith. Manuscript, 1989.

[11] F. Ergun, R. Kumar, and R. Rubinfeld, *Checking approximate computations of polynomials and functional equations*, SIAM J. Comput, 31 (2001), pp. 550–576.

[12] P. Gemmell, R. Lipton, R. Rubinfeld, M. Sudan, and A. Wigderson, *Self-testing/correcting for polynomials and for approximate functions*, in Proceedings of the Twenty-Third Annual ACM Symposium on Theory of Computing, 1991, pp. 32–42.

[13] J. Hastad, *Some optimal inapproximability results*, JACM, 48 (2001), pp. 798–859.

[14] J. Hastad and A. Wigderson, *Simple analysis of graph tests for linearity and pcp*, Random Structures and Algorithms, 22 (2003), pp. 139–160.

[15] C. Jutla, A. Patthak, A. Rudra, and D. Zuckerman, *Testing low-degree polynomials over prime fields.*, in Proceedings of the Forty-Fifth Annual Symposium on Foundations of Computer Science, 2004, pp. 423–432.

[16] T. Kaufman and S. Litsyn, *Almost orthogonal linear codes are locally testable*, in Proceedings of the Forty-Sixth Annual Symposium on Foundations of Computer Science, 2005, pp. 317–326.

[17] T. Kaufman, S. Litsyn, and N. Xie, *Breaking the ϵ-soundness bound of the linearity test over gf(2).* Private Communications, 2006.

[18] T. Kaufman and D. Ron, *Testing polynomials over general fields.*, in Proceedings of the Forty-Fifth Annual Symposium on Foundations of Computer Science, 2004, pp. 413–422.

[19] M. Kiwi, F. Magniez, and M. Santha, *Exact and approximate testing/correcting of algebraic functions: A survey*, Electronic Colloqium on Computational Complexity, 8 (2001).

[20] ——, *Approximate testing with error relative to input size*, JCSS, 66 (2003), pp. 371–392.

[21] F. Magniez, *Multi-linearity self-testing with relative error*, Theory Comput. Syst., 38 (2005), pp. 573–591.

[22] A. Polischuk and D. Spielman, *Nearly linear-size holographic proofs*, in Proceedings of the Twenty-Sixth Annual ACM Symposium on the Theory of Computing, 1994, pp. 194–203.

[23] R. Raz and S. 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 the Theory of Computing, 1997, pp. 475–484.

[24] R. Rubinfeld, *On the robustness of functional equations*, SIAM J. Comput, 28 (1999), pp. 1972–1997.

[25] R. Rubinfeld and M. Sudan, *Robust characterization of polynomials with applications to program testing*, SIAM Journal on Computing, 25 (1996), pp. 252–271.

[26] A. Samorodnitsky and L. Trevisan, *A pcp characterization of np with optimal amortized query complexity*, in stoc00, 2000, pp. 191–199.

[27] ——, *Gowers uniformity, influence of variables, and pcps*, in stoc06, 2006, pp. 11–20.

[28] A. Shpilka and A. Wigderson, *Derandomizing homomorphism testing in general groups*, in Proceedings of the Thirty-Sixth Annual ACM Symposium on the Theory of Computing, 2004, pp. 427–435.

[29] L. Trevisan, *Recycling queries in pcps and in linearity tests*, in Proceedings of the Thirtieth Annual ACM Symposium on the Theory of Computing, 1998, pp. 299–308.