Electronic Colloquium on Computational Complexity, Report No. 133 (2016)

A Simple $\widetilde{O}(n)$ Non-Adaptive Tester for Unateness

Deeparnab Chakrabarty
Microsoft Research Bangalore
dechakr@microsoft.com

C. Seshadhri
University of California, Santa Cruz
sesh@ucsc.edu

Abstract

Khot and Shinkar (RANDOM, 2016) recently describe an adaptive, $O(n \log(n)/\varepsilon)$-query tester for unateness of Boolean functions $f : \{0,1\}^n \mapsto \{0,1\}$. In this note we describe a simple non-adaptive, $O(n \log(n/\varepsilon)/\varepsilon)$ -query tester for unateness for functions over the hypercube with any ordered range.

1 Introduction

Let f be a function $f : \{0,1\}^n \mapsto R$ defined over the Boolean hypercube where R is some ordered range. We use e_i to denote the unit vectors in $\{0,1\}^n$ that has 1 in the ith coordinate, and 0s at other coordinates. The *i-th partial derivate* at x is $f(x \oplus e_i) - f(x)$ if $x_i = 0$ and $f(x) - f(x \oplus e_i)$ is $x_i = 1$. This is denoted by the function $\partial_i f$. Note that a function is monotonically increasing (or simply monotone), if $\partial_i f(x) \geq 0$ for all $x \in \{0,1\}^n$.

Unateness is a generalization of monotonicity. A function is unate if in every coordinate it is either monotone or anti-monotone. More precisely, a function f is unate if for all $i \in [n]$, either $\partial_i f(x) \geq 0$ for all x, or $\partial_i f(x) \leq 0$ for all x. The problem of unateness testing was introduced by Goldreich et al. [GGL+00] in their seminal paper on testing monotonicity. For Boolean functions $f : \{0,1\}^n \to \{0,1\}$, [GGL+00] described a non-adaptive $O(n^{3/2}/\varepsilon)$-query tester (while for monotonicity of Boolean functions they described a non-adaptive $O(n/\varepsilon)$-query tester). To our knowledge this problem was not studied after this, till very recently Khot and Shinkar[KS16] gave the first improvement for Boolean functions by designing an adaptive $O(n \log(n)/\varepsilon)$-query tester. Our main theorem is the following.

Theorem 1.1. *Consider functions $f : \{0,1\}^n \mapsto R$, where R is an arbitrary ordered set. There exists an one-sided error, non-adaptive, $O((n/\varepsilon)\log(n/\varepsilon))$-time tester for unateness.*

Monotonicity testing has been extensively studied in the past two decades [GGL+00, DGL+99, EKK+00, HK03, HK08, BGJ+09, BCGSM12, CS14, CS13, CST14, CDST15, CDJS15, BB16]. We employ a previous result on testing derivative bounded properties by Chakrabarty et al. [CDJS15]. That paper provides general theorems about the testability of properties that are specified in terms of the partial derivative being bounded.

Definition 1.2. *Given an n-dimensional bit vector $\mathbf{b}$, call a function $f : \{0,1\}^n \to R$ $\mathbf{b}$-monotone if for all i with $\mathbf{b}_i = 0$ we have $\partial_i f(x) \geq 0$ for all x, and for all i with $\mathbf{b}_i = 1$ we have $\partial_i f(x) \leq 0$ for all x.*

ISSN 1433-8092

Note that **0**-monotonicity is simply the standard notion of monotonicity and **1**-monotonicity is the notion of anti-monotonicity. Also note that a function is unate iff it is **b**-monotone for some **b**.

The property of **b**-monotonicity is a *derivative-bounded property*, in the language of [CDJS15]. A dimension reduction theorem for derivative properties (Theorem 8 in Arxiv version of [CDJS15]), when instantiated for **b**-monotonicity, immediately implies the following theorem.

Theorem 1.3. *Fix bit vector* **b** *and function* $f : \{0,1\}^n \to R$. *Let* ε *denote the distance of* f *to* **b**-*monotonicity. Let* μ_i *be the fraction of points where* $\partial_i f$ *violates* **b**-*monotonicity, that is, the number of hypercube edges across dimension* i *which violate* **b**-*montonicity is* $\mu_i 2^n$. *Then* $\sum_{i=1}^n \mu_i \geq \varepsilon/4$.

The above theorem can also be obtained by observing that f is **b**-monotone (resp, ε-far from being **b**-monotone) iff the function $g(x) := f(x \oplus b)$ is monotone (resp, ε-far from being monotone). Furthermore, every hypercube edge that violates monotonicity for g also violates **b**-monotonicity for f. A previous result of the authors shows that if a function g is ε-far from being monotone, then it has $\varepsilon 2^{n-1}$ hypercube edges violating monotonicity [CS13].

2 The Tester

`Unate-test`(f, ε)

1. For $r = 1, 2, \dots, L := \lceil \log(8n/\varepsilon) \rceil$:
   Repeat $s_r = \lceil \frac{20n}{\varepsilon \cdot 2^r} \rceil$ times
   (a) Sample u.a.r. dimension i.
   (b) Sample a set R_i of $3 \cdot 2^r$ u.a.r. points in the hypercube and evaluate $\partial_i f$ at all these points.
   (c) If there is some $x \in R_i$ such that $\partial_i f(x) > 0$ **and** $y \in R_i$ such that $\partial_i f(y) < 0$, reject and abort.
2. Accept (since tester has not rejected so far)

It is evident that this is a non-adaptive, one-sided tester. Furthermore, the running time is $O((n/\varepsilon)\log(n/\varepsilon))$. It suffices to prove the following.

Theorem 2.1. *If* f *is* ε-*far from being unate,* `Unate-test` *rejects with probability at least* $2/3$.

Proof. For dimension i, let U_i be the set of points in $\{0,1\}^n$ where $\partial_i f(x) > 0$. Analogous, let D_i be the set of points where $\partial_i f(x) < 0$. The tester rejects iff it finds a triple i, x, y such that $x \in U_i$ and $y \in D_i$. Let $\mu_i := \min(|U_i|, |D_i|)/2^n$.

Define the n-dimensional bit vector **b** as follows: $\mathbf{b}_i = 0$ if $|U_i| > |D_i|$ and $\mathbf{b}_i = 1$ otherwise. Observe that the fraction of points where $\partial_i f$ violates **b**-monotonicity is precisely μ_i. Since f is ε-far from being unate, f is ε-far from being **b**-monotone. By Theorem 1.3, $\sum_i \mu_i \geq \varepsilon/4$.

For any integer $r \geq 1$, define $S_r := \{i \in [n] \ : \ \mu_i \in (1/2^r, 1/2^{r-1}]\}$.

Claim 2.2. $\sum_{r=1}^L |S_r|/2^r \geq \varepsilon/8$.

Proof. Observe that $\sum_{r>L} |S_r|/2^r \leq (\varepsilon/8n) \sum_r |S_r| = \varepsilon/8$ since $\sum_r |S_r| = n$. Also, $\sum_r |S_r|/2^r \geq \sum_i \mu_i \geq \varepsilon/4$ from Theorem 1.3. We subtract these bounds to prove the claim. □

Fix r. Let p_r be the probability that Step 1a, Step 1b, and Step 1c reject for this r. Then the probability that the tester rejects is

$$1 - \prod_{r=1}^{L}(1-p_r)^{s_r} \geq 1 - e^{-\sum_{r=1}^{L} p_r s_r} \tag{1}$$

We now lower bound p_r. The tester rejects iff the set R_i in Step 1b contains a point in U_i and in D_i. The probability that R_i *does not* contain a point in U_i or D_i is at most $(1-|U_i|/2^n)^{|R_i|} + (1-|D_i|/2^n)^{|R_i|} \leq 2(1-\mu_i)^{|R_i|}$. Note that if the sampled dimension i lies in S_r, then this probability is at most $2\,(1-1/2^r)^{3\cdot 2^r} < 1/6$. Therefore, we get $p_r > \frac{5}{6} \cdot \frac{|S_r|}{n}$. Since $s_r \geq \frac{20n}{\varepsilon \cdot 2^r}$, we get

$$\sum_{r=1}^{L} p_r s_r \geq \sum_{r=1}^{L} \frac{5}{6} \cdot \frac{|S_r|}{n} \cdot \frac{20n}{\varepsilon 2^r} \geq \frac{100}{6\varepsilon} \sum_{r=1}^{L} \frac{|S_r|}{2^r} \geq 2$$

where the second ineqauality follows from Claim 2.2. Substituting in (1), we get the theorem. □

References

[BB16] A. Belovs and E. Blais. A polynomial lower bound for testing monotonicity. In *Proceedings of the Symposium on Theory of Computing (STOC)*, pages 1021–1032, 2016. 1

[BCGSM12] J. Briët, S. Chakraborty, D. García-Soriano, and A. Matsliah. Monotonicity testing and shortest-path routing on the cube. *Combinatorica*, 32(1):35–53, 2012. 1

[BGJ$^+$09] A. Bhattacharyya, E. Grigorescu, K. Jung, S. Raskhodnikova, and D. Woodruff. Transitive-closure spanners. In *Proceedings of the 18th Annual Symposium on Discrete Algorithms (SODA)*, pages 531–540, 2009. 1

[CDJS15] D. Chakrabarty, K. Dixit, M. Jha, and C. Seshadhri. Property testing on product distributions: Optimal testers for bounded derivative properties. In *Proceedings of the Symposium on Discrete Algorithms*, 2015. 1, 2

[CDST15] X. Chen, A. De, R. A. Servedio, and L-Y. Tan. Boolean function monotonicity testing requires (almost) $O(n^{1/2})$ non-adaptive queries. In *Proceedings of the Symposium on Theory of Computing (STOC)*, pages 519–528, 2015. 1

[CS13] D. Chakrabarty and C. Seshadhri. Optimal bounds for monotonicity and Lipschitz testing over hypercubes and hypergrids. In *Symposium on Theory of Computing (STOC)*, pages 419–428, 2013. 1, 2

[CS14] D. Chakrabarty and C. Seshadhri. An o(n) monotonicity tester for boolean functions over the hypercube. *SIAM Journal on Computing*, 2014. 1

[CST14] X. Chen, R. A. Servedio, and L-Y. Tan. New algorithms and lower bounds for monotonicity testing. In *Proceedings of Foundations of Computer Science (FOCS)*, pages 286–295, 2014. 1

[DGL+99] Y. Dodis, O. Goldreich, E. Lehman, S. Raskhodnikova, D. Ron, and A. Samorodnitsky. Improved testing algorithms for monotonicity. *Proceedings of the 3rd International Workshop on Randomization and Approximation Techniques in Computer Science (RANDOM)*, pages 97–108, 1999. 1

[EKK+00] F. Ergun, S. Kannan, R. Kumar, R. Rubinfeld, and M. Viswanathan. Spot-checkers. *Journal of Computer Systems and Sciences (JCSS)*, 60(3):717–751, 2000. 1

[GGL+00] O. Goldreich, S. Goldwasser, E. Lehman, D. Ron, and A. Samordinsky. Testing monotonicity. *Combinatorica*, 20:301–337, 2000. 1

[HK03] S. Halevy and E. Kushilevitz. Distribution-free property testing. *Proceedings of the 7th International Workshop on Randomization and Approximation Techniques in Computer Science (RANDOM)*, pages 302–317, 2003. 1

[HK08] S. Halevy and E. Kushilevitz. Testing monotonicity over graph products. *Random Structures and Algorithms*, 33(1):44–67, 2008. 1

[KS16] S. Khot and I. Shinkar. An $\widetilde{O}(n)$ queries adaptive tester for unateness. In *RANDOM*, 2016. To appear, Arxiv report 1608.02451. 1

ECCC ISSN 1433-8092

http://eccc.hpi-web.de