Preface

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. Remarkably, this approximate decision is made by accessing a small portion of the data set.

Property Testing has been a subject of intensive research in the last couple of decades, with hundreds of studies conducted in it and in closely related areas. Indeed, Property Testing is closely related to Probabilistically Checkable Proofs (PCPs), and is related to Coding Theory, Combinatorics, Statistics, Computational Learning Theory, Computational Geometry, and more.

The current volume provides a taste of the area of Property Testing. It grew out of a mini-workshop on property testing that took place in January 2010 in the Institute for Computer Science (ITCS) at Tsinghua University (Beijing). The mini-workshop brought together a couple of dozen of leading researchers in Property Testing and related areas. At the end of this mini-workshop it was decided to compile a collection of extended abstracts and surveys that reflects the program of the mini-workshop.

Property testing at a glance

Property testing is a relaxation of decision problems and it focuses on algorithms that can only read parts of the input. Thus, the input is represented as a function (to which the tester has oracle access) and the tester is required to accept functions that have some predetermined property (i.e., reside in some predetermined set) and reject any function that is "far" from the set of functions having the property. Distances between functions are defined as the fraction of the domain on which the functions disagree, and the threshold determining what is considered far is presented as a proximity parameter, which is explicitly given to the tester.

An asymptotic analysis is enabled by considering an infinite sequence of domains, functions, and properties. That is, for any n, we consider functions from D_n to R_n, where $|D_n| = n$. (Often, one just assumes that $D_n = [n] \stackrel{\rm def}{=} \{1, 2, ..., n\}$.) Thus, in addition to the input oracle, representing a function $f : D_n \to R_n$, the tester is explicitly given two parameters: a size parameter, denoted n, and a proximity parameter, denoted ϵ.

> **Definition:** *Let $\Pi = \bigcup_{n\in\mathbb{N}} \Pi_n$, where Π_n contains functions defined over the domain D_n. A* tester for a property *Π is a probabilistic oracle machine T that satisfies the following two conditions:*
>
> 1. *The tester accepts each $f \in \Pi$ with probability at least $2/3$; that is, for every $n \in \mathbb{N}$ and $f \in \Pi_n$* (and every $\epsilon > 0$)*, it holds that* $\Pr[T^f(n, \epsilon)=1] \geq 2/3$.

2. *Given $\epsilon > 0$ and oracle access to any f that is ϵ-far from Π, the tester rejects with probability at least 2/3; that is, for every $\epsilon > 0$ and $n \in \mathbb{N}$, if $f : D_n \to R_n$ is ϵ-far from Π_n, then $\Pr[T^f(n, \epsilon) = 0] \geq 2/3$, where f is ϵ-*far from *Π_n if, for every $g \in \Pi_n$, it holds that $|\{e \in D_n : f(e) \neq g(e)\}| > \epsilon \cdot n$.*

If the tester accepts every function in Π with probability 1, then we say that it has one-sided error*; that is, T has one-sided error if for every $f \in \Pi$ and every $\epsilon > 0$, it holds that $\Pr[T^f(n, \epsilon) = 1] = 1$. A tester is called* non-adaptive *if it determines all its queries based solely on its internal coin tosses* (and the parameters n and ϵ)*; otherwise it is called* adaptive*.*

This definition does not specify the query complexity of the tester, and indeed an oracle machine that queries the entire domain of the function qualifies as a tester (with zero error probability...). Needless to say, we are interested in testers that have significantly lower query complexity.

Research in property testing is often categorized according to the type of functions and properties being considered. In particular, algebraic property testing focuses on the case that the domain and range are associated with some algebraic structures (e.g., groups, fields, and vector spaces) and studies algebraic properties such as being a polynomial of low degree. In the context of testing graph properties, the functions represent graphs or rather allow certain queries to such graphs; for example, in the adjacency matrix model, graphs are represented by their adjacency relation and queries correspond to pairs of vertices where the answers indicate whether or not the two vertices are adjacent in the graph. (In an alternative model, known as the incidence-list model, graphs are represented by functions that assign to the pair (v, i) the ith neighbor of vertex v.)

Current research in property testing focuses mainly on query (and/or sample) complexity, while either ignoring time complexity or considering it a secondary issue. The current focus on these information theoretic measures is justified by the fact that even the latter are far from being understood. (Indeed, this stands in contrast to the situation in, say, PAC learning.)

The representation of problems' instances is crucial to any study of computation, since the representation determines the type of information that is explicit in the input. This issue becomes much more acute when one is only allowed partial access to the input (i.e., making a number of queries that result in answers that do not fully determine the input). An additional issue, which is unique to property testing, is that the representation may effect the distance measure (i.e., the definition of distances between inputs). This is crucial because property testing problems are defined in terms of this distance measure.

The contents of this volume

This volume contains extended abstracts of almost all works presented at the workshop as well as a large number of surveys. The surveys refer to various subareas of property testing and/or to research directions in property testing. Some of these surveys correspond to presentation that took place in the workshop, and others were written for this volume by some of the workshop's participants. The list of surveys include

- Eli Ben-Sasson: Limiting the Rate of Locally Testable Codes.
- Eric Blais: Testing juntas.
- Artur Czumaj and Christian Sohler: Sublinear-Time Algorithms.
- Oded Goldreich: A Brief Introduction to Propoerty Testing.
- Oded Goldreich: Locally Testing Codes and Proofs.
- Oded Goldreich: Testing Graph Properties.
- Ilan Newman: Property Testing in the "Massively Parameterized" Model.
- Krzysztof Onak: Sublinear Graph Approximation Algorithms.
- Sofya Raskhodnikova: Transitive-Closure Spanners.
- Rocco Servedio: Testing by Implicit Learning.
- Madhu Sudan: Invariance in Property Testing.

The list of extended abstracts includes

- Noga Alon: On fast approximation of graph parameters.
- Victor Chen: Testing Linear-Invariant Non-Linear Properties.
- Victor Chen: A Hypergraph Dictatorship Test with Perfect Completeness.
- Artur Czumaj: Testing Monotone Continuous Distributions on Real Cubes.
- Oded Goldreich: Algorithmic Aspects of Property Testing in the Dense Graphs Model.
- Prahladh Harsha: Composition of Low-Error 2-Query PCPs.
- Tali Kaufman: Symmetric LDPC Codes and Local Testing.
- Swastik Kopparty: Optimal Testing of Reed-Muller Codes.
- Michael Krivelevich: Comparing the Strength of Query Types.
- Michael Krivelevich: Hierarchy Theorems for Property Testing
- Kevin Matulef: Testing (subclasses of) Linear Threshold Functions.
- Krzysztof Onak: External Sampling.
- Krzysztof Onak: The Query Complexity of Edit Distance.
- Ronitt Rubinfeld: Maintaining a Large Matching or a Small Vertex Cover.
- Michael Saks: Local Monotonicity Reconstruction.
- Shubhangi Saraf: Some Recent Results on Testing of Sparse Linear Codes.
- Asaf Shapira: Testing Linear Invariant Properties.
- Christian Sohler: Testing Euclidean Spanners.

The surveys and extended abstracts appearing in this volume were not refereed. The extended abstracts refers to papers that have either appeared or are likely to appear in peer-reviewed conferences and journals.

Acknowledgments

I wish to thank all the authors who have contributed to the current volume as well as all researchers who have contributed to the research being surveyed in it.

Oded Goldreich
Weizmann Institute of Science

Table of Contents