Recent Progress in the Classification for Testability

Charles Jordan* and Thomas Zeugmann†
Department of Information Science and Technology
Hokkaido University, Sapporo 060-0814, Japan
* Tel/Fax: +81-11-706-7675, Email: skip@ist.hokudai.ac.jp
† Tel/Fax: +81-11-706-7684, Email: thomas@ist.hokudai.ac.jp

***Abstract*—In property testing we wish to distinguish between objects that have a given property and objects that are *far* from having the property, after examining only a small portion of the object. The classification problem for testability focuses on properties that are expressible in sentences of first-order logic, and asks us to determine which prefix vocabulary classes of first-order logic are testable and which are not.**

Here we review recent results concerning this classification.

I. Introduction

In property testing, we take a small, random sample of a large structure and wish to determine if the structure has some desired property or if it is far from having the property. The hope is that we can gain efficiency in return for not deciding the problem exactly. We are particularly interested in first-order expressible properties and seek to obtain a complete classification of the testable and untestable prefix vocabulary classes of first-order logic. Here we introduce some of our recent results concerning this classification; proofs of all theorems can be found in the references.

The paper is structured as follows. In Section II we briefly outline the history of property testing, focusing particularly on previous work concerning the classification for testability. We give definitions and notation in Section III. In Section IV we describe some recent results in the classification for testability.

II. Background

The study of property testing began in formal verification (see the recent introduction to graph property testing by Goldreich [1] and the recent surveys by Ron [2], [3] for references and additional background), and attention soon focused on testing graph and string properties. Alon *et al.* [4] showed that the regular languages are testable, implying the testability of monadic first-order logic[1].

Alon *et al.* [6] began a *logical* characterization of the testable (graph) properties. They showed that all graph properties expressible in first-order formulas with quantifier pattern "$\exists^*\forall^*$" (Ramsey's class) are testable, and that there exist untestable graph properties expressible with quantifier pattern "$\forall^*\exists^*$". The example they give is expressible with 17 variables and prefix $\forall^{12}\exists^5$. See Section IV for more recent results concerning the logical classification for testability.

The first author is supported by a Grant-in-Aid for JSPS Fellows under Grant No. 2100195209. The second author is supported by MEXT Grant-in-Aid for Scientific Research on Priority Areas under Grant No. 21013001.

[1]Recall that monadic first-order logic characterizes a subclass of the regular languages (cf. McNaughton and Papert [5]).

Although much of the research in property testing has focused on graph properties (see Alon and Shapira [7] for a survey), very recently there has been a significant effort to generalize these results in a variety of ways, usually via applications of generalizations of Szemerédi's Regularity Lemma. In particular, Austin and Tao [8] have recently shown that all hereditary properties of colored, directed hypergraphs are testable, generalizing a series of related results. We omit more detailed citations due to space constraints.

III. Preliminaries

We are interested in properties expressible in first-order logic, and so we begin with logical definitions. We restrict ourselves to very brief, informal definitions due to space constraints; see Jordan and Zeugmann [9] for details.

Definition 1. *A* vocabulary τ *is a tuple of distinct predicate symbols* R_i *together with their arities* a_i,

$$\tau := (R_1^{a_1}, \ldots, R_s^{a_s}) .$$

For example, the vocabulary of directed graphs is

$$\tau_G := (E^2).$$

Definition 2. *A* structure A of type τ *is an* $(s+1)$*-tuple*

$$A := (U, \mathcal{R}_1^A, \ldots, \mathcal{R}_s^A) ,$$

where U *is a finite universe and each* $\mathcal{R}_i^A \subseteq U^{a_i}$ *is a predicate corresponding to the predicate symbol* R_i *of* τ.

A *property* of type τ is any set of structures of type τ. In property testing, our goal is to distinguish between structures that have a property and those which are far from having the property. This requires a distance measure, for example we could say that a structure is ε-far from a property P if we must modify more than an ε-fraction of its representation to obtain a structure with P. See Jordan and Zeugmann [9] for a discussion of alternative distances.

Definition 3. *A property* P *is* testable *if, for every* $\varepsilon > 0$, *there exist randomized approximation algorithms satisfying the following. After examining at most* $c(\varepsilon)$ *bits of a structure* A, *the algorithm accepts with probability* $2/3$ *if* $A \in P$ *and rejects with probability* $2/3$ *if* A *is* ε*-far from* P.

Definition 3 is non-uniform in that we allow different algorithms for each choice of ε and so it is natural to consider uniformity conditions. There exist properties that are non-uniformly testable but not uniformly testable, however our results hold in both the uniform and non-uniform cases and so we will no longer distinguish between the two.

Finally, we briefly outline the relevant definitions for classification, see Börger *et al.* [10] for more formal definitions and history. A *prefix vocabulary class* is a triple $[\Pi, p]_e$, where Π is a string over the alphabet $\{\exists, \forall, \exists^*, \forall^*\}$, p is either a sequence over the naturals and first infinite ordinal or the phrase "*all*", and e is either '$=$' or the empty string.

Definition 4. *First-order formula $\varphi := \pi_1 x_1 \pi_2 x_2 \dots \pi_r x_r : \psi$ in prenex normal form, with quantifiers π_i and quantifier-free ψ, belongs to the* prefix vocabulary fragment *defined by $[\Pi, p]_e$ if the following conditions are satisfied.*

1) *The string $\pi_1 \pi_2 \dots \pi_r$ is contained in the language specified by Π when Π is interpreted as a regular expression.*
2) *If p is not* all*, at most p_i distinct predicate symbols of arity i appear in ψ.*
3) *Equality ($=$) appears in ψ only if e is '$=$'.*

The goal in the classification problem for testability is to obtain a complete classification of which prefix vocabulary classes are testable and which are not.

IV. Recent Results

The classification of prefix vocabulary classes of first-order logic was started by Alon *et al.* [6]. They showed that undirected, loop-free properties expressible in $[\exists^*\forall^*, (0,1)]_=$ are testable and that there exists an untestable property expressible in $[\forall^{12}\exists^5, (0,1)]_=$. When studying the classification problem, it is necessary to minimize the number of quantifiers needed to express untestable properties. Additionally, the first-order theory of graphs is not restricted to undirected, loop-free graphs. Accordingly, we have recently simplified the untestable example from Alon *et al.* [6], giving the following.

Theorem 1 (Jordan and Zeugmann [11])**.** *The following prefix vocabulary classes are not testable:*

1) $[\forall^3\exists, (0,1)]_=$
2) $[\forall^2\exists\forall, (0,1)]_=$
3) $[\forall\exists\forall^2, (0,1)]_=$
4) $[\forall\exists\forall\exists, (0,1)]_=$

The specific example used is essentially an encoding in directed graphs of undirected graph isomorphism[2]. Very recently, we have used a somewhat more complicated property (related to Boolean function isomorphism) to sharpen three of these prefixes as follows.

Theorem 2 (Jordan and Zeugmann [13])**.** *There are untestable properties (even given $o(\sqrt{n})$ queries) in $[\forall\exists\forall, (0,1)]_=$.*

The proof applies ideas from Alon and Blais [14]. However, first-order logic is not restricted to expressing properties of graphs and so it is natural to consider other types of structures. We generalized the concept of testability to relational structures and showed the following.

Theorem 3 (Jordan and Zeugmann [9])**.** *All formulas in $[\exists^*\forall\exists^*, all]_=$ define properties that are testable.*

This is Ackermann's class with equality, which has many nice properties. We have also applied a strong result by Austin and Tao [8] to extend the positive result of Alon *et al.* [6] from undirected, loop-free graphs to relational structures.

Theorem 4 (Jordan and Zeugmann [15])**.** *All formulas in $[\exists^*\forall^*, all]_=$ define properties that are testable.*

This is (the full) Ramsey's class, which also has many nice properties. The current classification is consistent with the classifications for the finite model property, docility and associated second-order 0-1 laws. We are particularly interested in the testability of variants of the Gödel class (i.e., classes containing $[\forall^2\exists, (0,1)]$). Determining whether these classes are testable would complete the classification for the special case of predicate logic with equality.

[2]Graph isomorphism is generally hard for testing, see, e.g., Fischer and Matsliah [12].

References

[1] O. Goldreich, "Introduction to testing graph properties," Electronic Colloquium on Computational Complexity (ECCC), Tech. Rep. TR10-082, May 2010.

[2] D. Ron, "Property testing: A learning theory perspective," *Found. Trends Mach. Learn.*, vol. 1, no. 3, pp. 307–402, 2008.

[3] ——, "Algorithmic and analysis techniques in property testing," *Found. Trends Theor. Comput. Sci.*, vol. 5, no. 2, pp. 73–205, 2009.

[4] N. Alon, M. Krivelevich, I. Newman, and M. Szegedy, "Regular languages are testable with a constant number of queries," *SIAM J. Comput.*, vol. 30, no. 6, pp. 1842–1862, 2001.

[5] R. McNaughton and S. Papert, *Counter-Free Automata*. M.I.T. Press, 1971.

[6] N. Alon, E. Fischer, M. Krivelevich, and M. Szegedy, "Efficient testing of large graphs," *Combinatorica*, vol. 20, no. 4, pp. 451–476, 2000.

[7] N. Alon and A. Shapira, "Homomorphisms in graph property testing," in *Topics in Discrete Mathematics*, ser. Algorithms and Combinatorics, M. Klazar, J. Kratochvíl, M. Loebl, J. Matoušek, R. Thomas, and P. Valtr, Eds. Springer, 2006, vol. 26, pp. 281–313.

[8] T. Austin and T. Tao, "On the testability and repair of hereditary hypergraph properties," *Random Struct. Algorithms*, vol. 36, no. 4, pp. 373–463, 2010.

[9] C. Jordan and T. Zeugmann, "Relational properties expressible with one universal quantifier are testable," in *Proc. SAGA 2009*, ser. Lecture Notes in Computer Science, O. Watanabe and T. Zeugmann, Eds., vol. 5792. Springer, 2009, pp. 141–155.

[10] E. Börger, E. Grädel, and Y. Gurevich, *The Classical Decision Problem*. Springer-Verlag, 1997.

[11] C. Jordan and T. Zeugmann, "Untestable properties expressible with four first-order quantifiers," in *Proc. LATA 2010*, ser. Lecture Notes in Computer Science, A.-H. Dediu, H. Fernau, and C. Martín-Vide, Eds., vol. 6031. Springer, 2010, pp. 333–343.

[12] E. Fischer and A. Matsliah, "Testing graph isomorphism," *SIAM J. Comput.*, vol. 38, no. 1, pp. 207–225, 2008.

[13] C. Jordan and T. Zeugmann, "Untestable properties in the Kahr-Moore-Wang class," in *Proc. WoLLIC 2011*, ser. Lecture Notes in Computer Science, L. Beklemishev and R. de Queiroz, Eds., vol. 6642. Springer, 2011, pp. 176–186.

[14] N. Alon and E. Blais, "Testing Boolean function isomorphism," in *Proc. RANDOM 2010*, ser. Lecture Notes in Computer Science, vol. 6302. Springer, 2010, pp. 394–405.

[15] C. Jordan and T. Zeugmann, "A note on the testability of Ramsey's class," in *Proc. TAMC 2010*, ser. Lecture Notes in Computer Science, J. Kratochvíl, A. Li, J. Fiala, and P. Kolman, Eds., vol. 6108. Springer, 2010, pp. 296–307.