The Complexity of Random Ordered Structures

Joel H. Spencer [1,2]

Courant Institute
New York University
251 Mercer Street
New York, NY 10012

Katherine St. John [1,3]

Department of Mathematics & Computer Science
Graduate Center & Lehman College
City University of New York
Bronx, NY 10468

Abstract

We show that for random bit strings, $U_p(n)$, with probability, $p = \frac{1}{2}$, the first-order quantifier depth $D(U_p(n))$ needed to distinguish non-isomorphic structures is $\Theta(\lg \lg n)$, with high probability. Further, we show that, with high probability, for random ordered graphs, $G_{\leq,p}(n)$ with edge probabiltiy $p = \frac{1}{2}$, $D(G_{\leq,p}(n)) = \Theta(\log^* n)$, contrasting with the results of random (non-ordered) graphs, $G_p(n)$, by Kim *et al.* [5] of $D(G_p(n)) = \log_{1/p} n + O(\lg \lg n)$.

Key words: random graphs, random bit strings, first order logic, Ehrenfeucht-Fraisse games

1 Introduction

Several natural ways exist for measuring the complexity of a structure: the number of variables used, the length of the sentence, and the "depth" of the

[1] We thank the referees for their helpful comments. The second author gratefully acknowledges support from NSF ITR 01-21651 and MRI 02-15942 and the hospitality of the Centre de Recerca Matemàtica, Barcelona, Spain.

[2] Email: `spencer@cs.nyu.edu`
[3] Email: `stjohn@lehman.cuny.edu`

quantifiers. We will focus on this last measure– the depth or the amount of nesting of the quantifiers in the sentence. The quantifier depth of the sentence corresponds to the number of registers needed to implement the sentence as a program and also to the number of moves in the Ehrenfeucht-Fraisse game [9].

Following [5], we define $D(\phi)$ to be the *quantifier depth* of the first order sentence ϕ and $D(G)$ to be the smallest depth of a sentence that defines the finite structure G. Kim *et al* [5] explore $D(G_p(n))$ for random (non-ordered) graphs, $G_p(n)$ and shows that for constant p, $D(G_p(n)) = O(\log n)$, with high probability, and for well-chosen values of (non-constant) p, that $D(G_p(n))$ can be $\Theta(\log^* n)$. The first order complexity of the random graph has also been studied in terms of convergence laws. Fagin and Glebski'i *et al.* [4,11] independently showed that random graphs, $G_p(n)$ with constant edge probability p, have a "zero-one" law. That is, for every first order sentence ϕ,

$$\lim_{n\to\infty} G_p(n) \models \phi = 0 \text{ or } 1$$

Shelah and Spencer [8] showed that for edge probability $p = n^{-\alpha}$, $0 < \alpha < 1$ the convergence depends on the value of α. For rational α, a zero-one law exists, but for irrational α, there exists first order sentences for which the above limit does not converge. Many stronger logics over random structures have been explored (see [2,9] for examples).

Another natural way to increase the expressive power is to add an ordering to the signature (see Chapter 11 of [9] for examples). In this paper, we focus on two such classes of structures: ordered random graphs and random bit strings. Surprisingly, for random graphs, we show that ordering not only gives a much smaller value for the $D(G_{\leq,p})$ for constant p, with high probability, but gives the lowest possible bound. That is, the quantifier depth needed to distinguish the random ordered graph with high probability is $D(G_{\leq,p}) = \Theta(\log^* n)$.

We also examine the natural class of ordered structures, that of random bit strings (also known as random ordered unary predicates). Zero-one laws have also been shown for random bit strings [1,6]. Spencer and St. John [10] examine the convergence rate for Zero-One laws and define the *tenacity* of a class of structures to capture the similarity when viewed via first order logic. In a general setting (a random structure defined for all n) and fixing positive ϵ, the *tenacity function*, $T_\epsilon(n)$, is equal to the maximal k so that if $n_1, n_2 \geq n$, then Duplicator wins this k-move Ehrenfeucht-Fraisse game (defined in § 2 below) played on independent structures of size n_1 and n_2 with probability at least $1 - \epsilon$. Spencer and St. John give bounds on the size of the tenacity function for several non-constant choices of the probability p. The tenacity is close related to the measure D. We show a tight bound for the quantifier depth $D(U_p)$ needed to distinguish random bit strings with probability $p = \frac{1}{2}$. Namely, $D(U_p) = \Theta(\lg \lg n)$.

In Section 2, we give some background and review past work. Section 3

contains the results for random bit strings. Section 4 contains the results for random ordered graphs. We conclude with open problems and future work.

2 Background

This section contains background information on games and probability. The expert may wish to skip the first two subsections and focus on the definitions in the last sections. For details about first order logic, see [3] for an excellent overview. For a more thorough treatment of games, probability, and logic, see [9].

2.1 The Ehrenfeucht-Fraisse Game

The game and its equivalence are due to Ehrenfeucht and Fraisse, and the presentation here is from [9].

In the Ehrenfeucht-Fraisse Game, the players alternate placing pebbles on one of two structures that serve as the game boards. The number of rounds that are played correspond to the complexity of first order sentences considered. Given two structures, $\mathcal{M}_1$ and $\mathcal{M}_2$, $\mathcal{M}_1$ and $\mathcal{M}_2$ are indistinguishable by first order sentences with quantifier rank at most k (written $\mathcal{M}_1 \equiv_k \mathcal{M}_2$) if and only if the second player has a winning strategy for every k-pebble Ehrenfeucht-Fraisse game of finite number of moves played on $\mathcal{M}_1$ and $\mathcal{M}_2$. We define the game below:

Definition 2.1 *The k-**pebble Ehrenfeucht-Fraisse game (EF game)** on $\mathcal{M}_1$ and $\mathcal{M}_2$ is a two-person game of perfect information. For the game, we have:*

- **Players:** *There are two players:*
  - *Player I, often called* Spoiler*, who tries to ruin any correspondence between the structures.*
  - *Player II, often called* Duplicator*, who tries to duplicate Spoiler's last move.*
- **Equipment:** *We have k pairs of pebbles and the two structures $\mathcal{M}_1$ and $\mathcal{M}_2$ as game boards.*
- **Moves:** *The players take turns moving. At the ith move, Spoiler chooses a structure and places his ith pebble on an element in that structure. Duplicator then places her corresponding pebble on an element in the other structure.*
- **Winning:** *If after any of Duplicator's moves, the substructures induced by the pebbles are not isomorphic, then Spoiler wins. After both players have played k moves, if Spoiler has not won, then Duplicator wins.*

2.2 Random Ordered Structures

Following [10], we define the *random bit string* or random unary predicate as follows. The signature is $\sigma = \{U, \leq\}$, where U is an unary predicate and $\leq$ is a binary predicate that follows the axioms of linear order. Let n be a positive integer, and $0 \leq p(n) \leq 1$. We will write $U(i)$ if the ith bit is "on" in the string, and $\neg U(i)$ if the ith bit is "off." U is an unary predicate– there are 2^n possible choices for such a predicate over n elements. The random bit string $U_p(n)$ is a probability space over predicates U on $[n] = \{1, \ldots, n\}$ with the probabilities determined by $\Pr[U(x)] = p(n)$, for $1 \leq x \leq n$, and the events $U(x)$ are mutually independent over $1 \leq x \leq n$.

The *random ordered graph* is defined similarly (see [9]). The signature is $\sigma = \{\sim, \leq\}$, where $\sim$ is a binary predicate that will represent edges and $\leq$ is a binary predicate that follows the axioms of linear order. Let n be a positive integer, and $0 \leq p(n) \leq 1$. We will write $i \sim j$ if there is an edge between i and j. The random ordered graph $G_{\leq,p}(n)$ is a probability space over all ordered graphs on n vertices with the edge probabilities determined by $\Pr[i \sim j] = p(n)$, for $1 \leq i, j \leq n$, and the events $i \sim j$ are mutually independent and uniformly distributed.

2.3 Complexity of First Order Structures

We measure the complexity of a first order sentence ϕ by its quantifier depth, which is the longest sequence of embedded quantifiers in ϕ. For example, a complete graph on n vertices can be described by the first order sentence:

$$A : \forall x \forall y (x \sim y) \wedge \exists x_1 \cdots x_n (\bigwedge_{i<j} x_i \neq x_j \wedge \forall y (\bigvee_i y = x_i)$$

where we write "$x \sim y$" when there is an edge between x and y. The first part of the sentence says that there is an edge between every pair of points. The second part says there is at least n distinct points. This sentence has quantifier depth $D(\phi) = n + 1$. Note that any finite graph can be described by a first order sentence, possibly very large in size, by listing every vertex and describing the edges that exist, and do not exist, in terms of the specific vertices. It is often possible to do much better than this as shown above by the sentence describing the complete graph on n vertices.

With this in mind, we focus on $D(G)$, as defined by Pikhurko *et al.* [7]:

$$D(G) = \min\{D(\phi) \mid G \models \phi \ \& \text{ if } H \not\cong G \text{ then } H \not\models \phi\}$$

The trivial bounds on this function, for any structure, are:

$$\Omega(\log^* n) \leq D(G) \leq O(\lg n)$$

where $\log^* n = \min\{i \in \mathbb{N} \mid \log_2^{(i)} n < 1\}$.

3 Random Bit Strings

We focus on the case where the probability is $p = \frac{1}{2}$ for the random bit string. As stated in Section 2.3, we have as a lower bound $D(G) = \Omega(\log^* n)$. We can improve this to $\lg \lg n$ by a straightforward argument on the occurrence of words in the random bit string.

Lemma 3.1 *If y is a substring of x, then $D(y) \leq D(x)$.*

Proof: If y is a substring of x, then it can be written as $x = pys$ where p (prefix) and s (suffix) are some binary strings. If $D(y) = k$, then there is another y' so that Duplicator wins the $k-1$ move game on y, y'. But then Duplicator wins the $k-1$ move game on $pys, py's$ by playing identically whenever Spoiler plays in the prefix or suffix. Hence $D(x) \geq k$. □

We use this lemma to show our first theorem– a lower bound on the quantifier depth for bit strings. The idea of the proof is that if a bit string U of length n contains a particular substring, for example 0^L, and another bit string U' contains the slightly different substring, say 0^{L+1}, then it will take Spoiler at least $\Omega(\lg L)$ moves to distinguish the two structures. When $L = \lg n$, words of length L occur with high probability, so, it will take Spoiler at least $\Omega(\lg L) = \Omega(\lg \lg n)$ to distinguish the structures. This corresponds to the lower bound $D(U_p(n)) = \Omega(\lg \lg n)$ in the theorem below.

Theorem 3.2 *For the random bit string, $U_p(n)$, with $p = \frac{1}{2}$, $D(U_p(n)) = \Omega(\lg \lg n)$ with high probability.*

Proof: With the above lemma, the lower bound follows from showing that

(i) With high probability, the random n-length bit string contains 0^L for $L = \lfloor 0.9 \lg n \rfloor$.
(ii) $D(0^L) \geq \lg L$.

For the first part, we split the n-length bit string into $\lfloor n/L \rfloor$ disjoint strings of length L (plus some excess). Each such string is O^L with probability 2^{-L} so the probability that none of them is 0^L is $(1 - 2^{-L})^{\lfloor n/L \rfloor} \to 0$.

For the second part, we need to show $D(0^L) \geq \lg L$. To show this, we play the Ehrenfeucht-Fraisse game on 0^L and 0^{L+1}. By Theorem 2.6.3 (p. 41 of [9]), Duplicator has a winning strategy for k move game on two totally ordered sets on n, m elements if and only if $n = m$ or $m, n \geq 2^k - 1$. So, for $k = \lg L$, Duplicator wins the k move game and $D(U) \geq \lg L$. By the lemma, $D(U) \geq D(0^L)$, thus, $D(U) \geq \lg \lg n$. □

The upper bound is more complicated, and relies on the fact that for a suitable choice of L, each string of length L occurs at most once. The uniqueness of the strings of length L allows us to describe the structure in small quantifier depth.

Theorem 3.3 *For the random bit string, $U_p(n)$ with $p = \frac{1}{2}$, $D(U_p(n)) = O(\lg \lg n)$ with high probability.*

Proof: The proof relies on three parts:

(i) For any L, each string s of length L can be described in $\lg L$ quantifier depth.
(ii) With $L = \lfloor 2.1 \lg n \rfloor$, with high probability, no string of length L occurs more than once.
(iii) With parts (1) and (2), we can describe the structure with appropriately small quantifier depth.

Note that the first part is straightforward since any bit string of length n can be described completely by a sentence of quantifier depth $\lg n$. The idea is to follow a "divide-and-conquer" approach to specify each element and to say if the element is a 0 or 1 (see [2,9] for details).

For the second part, we let $L = \lfloor 2.1 \lg n \rfloor$ and show that no string of length L occurs more than once. For a fixed L, the expected number of strings s of length L that occur twice is $\leq \frac{(n-L)(n-L-1)}{2}(1/2)^L$. If $L = \lfloor 2.1 \lg n \rfloor$, the expected number of strings occurring twice goes to zero, with high probability.

For the last part, we need to completely describe $U_p(n)$ in $O(\lg \lg n)$ with high probability. Since each string of length L occurs at most once, with high probability, we can reduce the description of the structure to describing the order in which the strings of length L occur. As noted above, to describe an individual string of length L takes $O(\lg L)$ quantifier depth. To describe two subsequent occurrences of strings, s and t of length L takes $O(\lg L)$. This can be done by defining a predicate NEXT such that:

$$\text{NEXT}[S,T] : \exists x(\text{“}[x, x+L) \text{ is } S\text{”} \wedge \text{“}[x+1, x+L) \text{ is } T\text{”})$$

and the predicate INIT:

$$\text{INIT}[S] : (\text{“}[0, L) \text{ is } S\text{”})$$

Given the starting position x, it takes $\lfloor \lg L \rfloor + 2$ quantifier depth to describe each clause of NEXT. INIT can also be described in $\lfloor \lg L \rfloor + 2$ quantifier depth. For each of the 2^{2L} pairs (S,T) we would have either NEXT$[S,T]$ or $\neg$NEXT$[S,T]$. This combined with the nonduplication determines the sequence. Using the "divide-and-conquer" approach again, we can describe the entire structure in $\lg L$ quantifier depth. Thus, $D(U_p) = O(\lg L) = O(\lg \lg n)$, with high probability. □

4 Random Ordered Graphs

For random ordered graphs, we have the general lower bound of $\Omega(\log^* n)$ for any structure (see Section 2.3). We show that the upper bound matches the best possible lower bound. To show the upper bound is complicated and relies on a recursive argument for the size of the sentence that defines a given structure $\mathcal{G}$.

For clarity in the proofs, we separate the logic and the probability results. First, we give a general bound that if a graph G satisfies some adjacency conditions (listed below in Theorem 4.1), then $D(G)$ can be bounded nicely from above. We then show that the random ordered graph on n vertices, $G_{\leq,p}(n)$, satisfies the adjacency properties and can achieve the upper bound of $O(\log^* n)$ by recursively applying Theorem 4.1.

Theorem 4.1 *Let $a_0 < a_1 < \ldots < a_k = n$. Suppose that for each $0 \leq i < k$ the points in $(a_i, a_{i+1}]$ have distinct "profiles" in $[1, a_i]$ – that is, no distinct $x, y \in (a_i, a_{i+1}]$ have precisely the same adjacencies to $[1, a_i]$. Then $D(G) \leq a_0 + 2k + 4$.*

Proof: Let G be a graph on n vertices and let $a_0 < \ldots < a_k = n$ satisfy the hypotheses of the theorem. Suppose $G' \not\simeq G$. We give a strategy for Spoiler that wins the Ehrenfeucht-Fraisse game in (at most) $a_0 + 2k + 4$ rounds. We label vertices in both graphs with their ordinals given by their respective orderings. For $x \in [a_i, a_{i+1})$, $0 \leq i < k$, we let PRO$[x]$, the profile of x, denote the set of $y \in [1, a_i]$ which are adjacent to x.

Spoiler first plays $1, \ldots, a_0 - 1$ on G. Duplicator must respond with $0, 1, \ldots, a_0 - 1$ on G' because if her responses were not consecutive, Spoiler would play in the gap and win in one further move. At this stage, we are assured that $G|_{[0,a_0)} \simeq G'|_{[0,a'_0)}$ (since $a'_0 = a_0$). Spoiler then plays $a_k = n$ and Duplicator must play $a'_k = n'$, the last vertex in the ordering, as otherwise Spoiler would select an $x > a'_k$ and would win in one further move. Spoiler then plays $a_1, \ldots, a_{k-1}$ and Duplicator responds with some $a'_1, \ldots, a'_{k-1}$. For $x' \in G'$ with $x \in [a'_{i-1}, a'_i)$, $1 \leq i \leq k$, we let PRO$'[x']$, the profile of x', denote the set of $y < a'_{i-1}$ which are adjacent to x'.

Suppose G' had an i, $0 \leq i < k$ and distinct $y', z' \in [a'_i, a'_{i+1})$ with PRO$[y']$ = PRO$[z']$. Spoiler would then select $y', z' \in G'$ and Duplicator would need select distinct $y, z \in [a_i, a_{i+1})$ in G. Our hypothesis on G insures that PRO$[y] \neq$ PRO$[z]$. Then Spoiler would select $x \in G$ with $x \in [0, a_i)$ such that precisely one of y, z are adjacent to x and Duplicator would have no response.

We call the above the initial phase. It lasts at most $a_0 + k + 3$ moves. Let us suppose that Duplicator has not already lost and call the remaining moves the final phase. We first have an auxilliary result for $0 \leq i \leq k$.

COPY$[i]$: Suppose $G|_{[0,a_i)} \simeq G'|_{[0,a'_i)}$ and $x \in G$, $x' \in G'$ with $x \neq x'$ and further suppose the first round of the final phase consist of the moves x, x'. Then Spoiler can win with a total (including the first round) of (at most) $i+1$ moves in the final phase.

For $i = 0$ there is nothing to show. Assume COPY$[i-1]$, and let x, x' satisfy the assumptions of COPY$[i]$. If $x, x' < a_i$ then $i - 1$ moves suffice by COPY$[i-1]$. As a_i, a'_i were played in the initial phase we must therefore have $x, x' > a_i$. But then PRO$[x] \neq$ PRO$[x']$. Spoiler selects $y < a_{i-1}$ in the symmetric difference of the sets. Duplicator cannot select the same y and so

must pick $y' < a'_{i-1}$ and now Spoiler wins in $i-1$ further rounds by induction.

Now we make our main result for $0 \leq i \leq k$.

IND[i]: If $G|_{[0,a_i)} \not\simeq G'|_{[0,a'_i)}$, then Spoiler can win with (at most) $i+1$ additional moves.

Our initial phase has disposed of the $i = 0$ case. Assume IND[$i-1$] and suppose $G|_{[0,a_i)} \not\simeq G'|_{[0,a'_i)}$. We may assume $G|_{[0,a_{i-1})} \simeq G'|_{[0,a'_{i-1})}$ as otherwise, by induction, Spoiler wins in only i moves. Suppose $\mathrm{PRO}[a_i] \neq \mathrm{PRO}'[a'_i]$. Recall a_i, a'_i were already selected in the initial phase. Spoiler would select $y < a_i$ in their symmetric difference, Duplicator would necessarily select $y' < a'_i$ with $y' \neq y$ and, by COPY[i], Spoiler would win in only $i+1$ total extra moves. Now suppose the pairs $(\mathrm{PRO}[x], \mathrm{PRO}[x+1])$, $a_{i-1} \leq x < a_i - 1$, and the pairs $(\mathrm{PRO}'[x'], \mathrm{PRO}'[x'+1])$, $a'_{i-1} \leq x' < a'_i - 1$ were not the same. Lets suppose, the other case being identical, that some $(\mathrm{PRO}[x], \mathrm{PRO}[x+1])$ was not a $(\mathrm{PRO}'[x'], \mathrm{PRO}'[x'+1])$. Spoiler selects $x, x+1$. Duplicator selects some $x', x'+1$. (If these are not consecutive Spoiler plays in the gap and wins immediately.) Now suppose, the other case being identical, that $\mathrm{PRO}[x] \neq \mathrm{PRO}'[x']$. Spoiler plays y in the symmetric difference (in either graph) and Duplicator must play $y' \neq y$. From COPY[i], Spoiler wins in $i-1$ further moves for a total of $i+1$ extra moves. Thus G' has the same initial profile and the same pairs of profiles as G and neither has duplicate profiles. They are isomorphic when restricted to $a_{i-1} = a'_{i-1}$, and hence they would be isomorphic up to $a_i = a'_i$.

Theorem 3 follows from IND[k]. □

We end this section by showing the probability argument for the upper bound.

Theorem 4.2 *For the random ordered graph, $G_{\leq,p}(n)$, with edge probability $p = \frac{1}{2}$, $D(G_{\leq,p}(n)) = \Theta(\log^* n)$.*

Proof: The lower bound comes from the general lower bound on all structures (see Section 2.3).

For the upper bound, we use Theorem 3. Note if the conditions hold for a graph G, then $D(G) \leq 2k + 4 + \lg a_0$.

Let $a_0 = \log^* n$ and $a_{i+1} = \lfloor 2^{a_i/4} \rfloor$ for $i > 0$. We first claim that $k = O(\log^* n)$ where $a_k = n$. This gives $D(G) = 2O(\log^* n)+4+\log^* n = O(\log^* n)$. To show the claim, we introduce some notation: let $b_0 = \log^* n$, and $b_{i+1} = 2^{b_i}$ for $i > 0$. Note that for $l = \log^* n$ that $b_l = \mathrm{Tower}(\log^* n) = n$. By induction (and some tedious technical details), $a_{4i} \geq b_i$ for $i > 0$. This gives $a_{4l} \geq b_l = n$. So, $k = \Theta(l) = \Theta(\log^* n)$.

Next, we need to show that the intervals (a_i, a_{i+1}) have unique profiles in $[1, a_i]$. Assume not. The probability for failure for a particular i is less then $\frac{(a_{i+1}-a_i)^2}{2} 2^{-a_i}$, which is bounded above by $a_{i+1}^2 \cdot 2^{-a_i}$. Since $a_{i+1} = \lfloor 2^{a_i/4} \rfloor$, this is less than $2^{-a_i/2}$. The total failure probability is less than $\sum_i 2^{-a_i/2} \leq \sum_{j \geq a_0} 2^{-j/2} = O(2^{-a_i/2}) = o(1)$ as a_0 goes to infinity. Since the hypothesis of the Theorem 3 are satisfied, we have the desired result. □

5 Conclusion and Future Work

We show tight bounds for the quantifier depth needed to distinguish structures for two natural classes of random ordered structures: bit strings and graphs. Our work focused on random structures with constant probability $p = \frac{1}{2}$. Related open questions include the complexity of other random ordered structures and the complexity of bit strings and graphs with non-constant probabilities.

References

[1] Peter Dolan. A zero-one law for a random subset. *Random Structures and Algorithms*, 2:317–326, 1991.

[2] Heinz-Dieter Ebbinghaus and Jorg Flum. *Finite Model Theory.* Springer, Berlin, 1995.

[3] Herbert B. Enderton. *A Mathematical Introduction to Logic.* Academic Press, San Diego, 1972.

[4] Ronald Fagin. Generalized first-order spectra and polynomial-time recognizable sets. In *Complexity of Computation, SIAM-AMS Proceedings*, pages 43–73, 1974.

[5] J. H. Kim, O. Pikhurko, J. Spencer, and O. Verbitsky. How complex are random graphs in first order logic? to appear in Random Structures and Algorithms, 2005.

[6] James F. Lynch. Probability of first-order sentences about unary functions. *Transactions of the American Mathematics Society*, 287:543–568, 1985.

[7] O. Pikhurko, H. Veith, and O. Verbitsky. The first order definability of graphs: Upper bounds for quantifier ranks. Submitted. E-print `arXiv:math.LO/0305244`, 2003.

[8] Saharon Shelah and Joel H. Spencer. Zero-one laws for sparse random graphs. *American Journal of Mathematics*, 1:95–102, 1988.

[9] Joel H. Spencer. *The Strange Logic of Random Graphs.* Springer, Berlin, 2001.

[10] Joel H. Spencer and Katherine St. John. The tenacity of zero-one laws. *The Electronic Journal of Combinatorics*, 8(2):R17, 2001.

[11] M.Liogon'kii Y. Glebski, D. Kogan and V. Talanov. Range and degree of realizability of formulas in the restricted predicate calculus. *Kibernetica*, 2:17–28, 1969. (English translation: Cybernetics **5**, (1969) 142-154.).