On the Readability of Overlap Digraphs

Rayan Chikhi[1,2], Paul Medvedev[2], Martin Milanič[3], and Sofya Raskhodnikova[2]

[1]CNRS, UMR 9189, France,
[2]The Pennsylvania State University, USA,
[3]University of Primorska, Slovenia

Abstract. We introduce the graph parameter *readability* and study it as a function of the number of vertices in a graph. Given a digraph D, an injective overlap labeling assigns a unique string to each vertex such that there is an arc from x to y if and only if x properly overlaps y. The readability of D is the minimum string length for which an injective overlap labeling exists. In applications that utilize overlap digraphs (e.g., in bioinformatics), readability reflects the length of the strings from which the overlap digraph is constructed. We study the asymptotic behaviour of readability by casting it in purely graph theoretic terms (without any reference to strings). We prove upper and lower bounds on readability for certain graph families and general graphs.

1 Introduction

In this paper, we introduce and study a graph parameter called readability, motivated by applications of overlap graphs in bioinformatics. A string x *overlaps* a string y if there is a suffix of x that is equal to a prefix of y. They overlap *properly* if, in addition, the suffix and prefix are both proper. The *overlap digraph* of a set of strings S is a digraph where each string is a vertex and there is an arc from x to y (possibly with $x = y$) if and only if x properly overlaps y. Walks in the overlap digraph of S represent strings that can be spelled by stitching strings of S together, using the overlaps between them. Overlap digraphs have various applications, e.g., they are used by approximation algorithms for the Shortest Superstring Problem [Swe00]. Their most impactful application, however, has been in bioinformatics. Their variants, such as de Bruijn graphs [IW95] and string graphs [Mye05], have formed the basis of nearly all genome assemblers used today (see [MKS10,NP13] for a survey), successful despite results showing that assembly is a hard problem in theory [BBT13,NP09,MGMB07]. In this context, the strings of S represent known fragments of the genome (called *reads*), and the genome is represented by walks in the overlap digraph of S. However, do the overlap digraphs generated in this way capture all possible digraphs, or do they have any properties or structure that can be exploited?

Braga and Meidanis [BM02] showed that overlap digraphs capture all possible digraphs, i.e., for every digraph D, there exists a set of strings S such that their overlap digraph is D. Their proof takes an arbitrary digraph and shows how to construct an *injective overlap labeling*, that is, a function assigning a unique

string to each vertex, such that (x, y) is an arc if and only if the string assigned to x properly overlaps the string assigned to y. However, the *length* of strings produced by their method can be exponential in the number of vertices. In the bioinformatics context, this is unrealistic, as the read size is typically much smaller than the number of reads.

To investigate the relationship between the string length and the number of vertices, we introduce a graph parameter called *readability*. The readability of a digraph D, denoted $r(D)$, is the smallest nonnegative integer r such that there exists an injective overlap labeling of D with strings of length r. The result by [BM02] shows that readability is well defined and is at most $2^{\Delta+1} - 1$, where Δ is the maximum of the in- and out-degrees of vertices in D. However, nothing else is known about the parameter, though there are papers that look at related notions [BFK$^+$02,BFKK02,BHKdW99,GP14,LZ07,LZ10,PSW03,TU88].

In this paper, we study the asymptotic behaviour of readability as a function of the number of vertices in a graph. We define readability for undirected bipartite graphs and show that the two definitions of readability are asymptotically equivalent. We capture readability using purely graph theoretic parameters (i.e., without any reference to strings). For trees, we give a parameter that characterizes readability exactly. For the larger family of bipartite C_4-free graphs, we give a parameter that approximates readability to within a factor of 2. Finally, for general bipartite graphs, we give a parameter that is bounded on the same sets of graphs as readability.

We apply our purely graph theoretic interpretation to prove readability upper and lower bounds on several graph families. We show, using a counting argument, that almost all digraphs and bipartite graphs have readability of at least $\Omega(n/\log n)$. Next, we construct a graph family inspired by Hadamard codes and prove that it has readability $\Omega(n)$. Finally, we show that the readability of trees is bounded from above by their radius, and there exist trees of arbitrary readability that achieve this bound.

2 Preliminaries

General definitions and notation. Let x be a string. We denote the length of x by $|x|$. We use $x[i]$ to refer to the i^{th} character of x, and denote by $x[i..j]$ the substring of x from the i^{th} to the j^{th} character, inclusive. We let $\text{pre}_i(x)$ denote the prefix $x[1..i]$ of x, and we let $\text{suf}_i(x)$ denote the suffix $x\,[|x| - i + 1..|x|]$. Let y be another string. We denote by $x \cdot y$ the concatenation of x and y. We say that x *overlaps* y if there exists an i with $1 \leq i \leq \min\{|x|, |y|\}$ such that $\text{suf}_i(x) = \text{pre}_i(y)$. In this case, we say that x overlaps y by i. If $i < \min\{|x|, |y|\}$, then we call the overlap *proper*. Define $\text{ov}(x, y)$ as the minimum i such that x overlaps y by i, or 0 if x does not overlap y. For a positive integer n, we denote by $[n]$ the set $\{1, \ldots, n\}$.

We refer to finite simple undirected graphs simply as graphs and to finite directed graphs without parallel arcs in the same direction as digraphs. For a vertex v in a graph, we denote the set of neighbors of v by $N(v)$. A *biclique* is a complete bipartite graph. Note that the one-vertex graph is a biclique (with one of the parts of its bipartition being empty). Two vertices u, v in a graph are

called *twins* if they have the same neighbors, i.e., if $N(u) = N(v)$. If, in addition, $N(u) = N(v) \neq \emptyset$, vertices u, v are called *non-isolated twins*. A *matching* is a graph of maximum degree at most 1, though we will sometimes slightly abuse the terminology and not distinguish between matchings and their edge sets. A cycle (respectively, path) on i vertices is denoted by C_i (respectively, P_i). For graph terms not defined here, see, e.g., [BM08].

Readability of digraphs. A *labeling* ℓ of a graph or digraph is a function assigning a string to each vertex such that all strings have the same length, denoted by $len(\ell)$. We define $\mathrm{ov}_\ell(u, v) = \mathrm{ov}(\ell(u), \ell(v))$. An *overlap labeling* of a digraph $D = (V, A)$ is a labeling ℓ such that $(u, v) \in A$ if and only if $0 < \mathrm{ov}_\ell(u, v)) < len(\ell)$. An overlap labeling is said to be *injective* if it does not generate duplicate strings. Recall that the readability of a digraph D, denoted $r(D)$, is the smallest nonnegative integer r such that there exists an injective overlap labeling of D of length r. We note that in our definition of readability we do not place any restrictions on the alphabet size. Braga and Meidanis [BM02] gave a reduction from an overlap labeling of length ℓ over an arbitrary alphabet Σ to an overlap labeling of length $\ell \log |\Sigma|$ over the binary alphabet.

Readability of bipartite graphs. We also define a modified notion of readability that applies to balanced bipartite graphs as opposed to digraphs. We found that readability on balanced bipartite graphs is simpler to study but is asymptotically equivalent to readability on digraphs. Let $G = (V, E)$ be a bipartite graph with a given bipartition of its vertex set $V(G) = V_s \cup V_p$. (We also use the notation $G = (V_s, V_p, E)$.) We say that G is *balanced* if $|V_s| = |V_p|$. An *overlap labeling of* G is a labeling ℓ of G such that for all $u \in V_s$ and $v \in V_p$, $(u, v) \in E$ if and only if $\mathrm{ov}_\ell(u, v) > 0$. In other words, overlaps are exclusively between the suffix of a string assigned to a vertex in V_s and the prefix of a string assigned to a vertex in V_p. The *readability* of G is the smallest nonnegative integer r such that there exists an overlap labeling of G of length r. Note that we do not require injectivity of the labeling, nor do we require the overlaps to be proper. As before, we use $r(G)$ to denote the readability of G.

We note that in our definition of readability we do not place any restrictions on the alphabet size. Braga and Meidanis [BM02] gave a reduction from an overlap labeling of length ℓ over an arbitrary alphabet Σ to an overlap labeling of length $\ell \log |\Sigma|$ over the binary alphabet.

For a labeling ℓ, we define $inner_i(\ell(v)) = \mathrm{suf}_i(\ell(v))$ if $v \in V_s$ and $inner_i(\ell(v)) = \mathrm{pre}_i(\ell(v))$ if $v \in V_p$. Similarly, we define $outer_i(\ell(v)) = \mathrm{pre}_i(\ell(v))$ if $v \in V_s$ and $outer_i(\ell(v)) = \mathrm{suf}_i(\ell(v))$ if $v \in V_p$.

Let $\mathcal{B}_{n\times n}$ be the set of balanced bipartite graphs with nodes $[n]$ in each part, and let $\mathcal{D}_n$ be the set of all digraphs with nodes $[n]$. The readabilities of digraphs and of bipartite graphs are connected by the following theorem, which implies that they are asymptotically equivalent.

Theorem 1. *There exists a bijection $\psi : \mathcal{B}_{n\times n} \to \mathcal{D}_n$ with the property that for all $G \in \mathcal{B}_{n\times n}$ and $D \in \mathcal{D}_n$, such that $D = \psi(G)$, we have that $r(G) < r(D) \leq 2 \cdot r(G) + 1$.*

As a result, we can study readability of balanced bipartite graphs, without asymptotically affecting our bounds. For example, we show in Section 4.2 (in Theorem 6) that there exists a family of balanced bipartite graphs with readability $\Omega(n)$, which leads to the existence of digraphs with readability $\Omega(n)$.

3 Graph theoretic characterizations

In this section, we relate readability of balanced bipartite graphs to several purely graph theoretic parameters, without reference to strings.

3.1 Trees and C_4-free graphs

For trees, we give an exact characterization of readability, while for C_4-free graphs, we give a parameter that is a 2-approximation to readability. A *decomposition of size* k of a bipartite graph $G = (V_s, V_p, E)$ is a function on the edges of the form $w : E \to [k]$. Note that a labeling ℓ of G implies a decomposition of G, defined by $w(e) = \text{ov}_\ell(e)$ for all $e \in E$. We call this the ℓ-decomposition. We say that a labeling ℓ of G *achieves* w if it is an overlap labeling and w is the ℓ-decomposition. Note that we can express readability as

$$r(G) = \min\{k \mid w \text{ is a decomposition of size } k\,, \exists \text{ a labeling } \ell \text{ that achieves } w\}\,.$$

Our goal is to characterize in graph theoretic terms the properties of w which are satisfied if and only if w is the ℓ-decomposition, for some ℓ. While this is challenging in general, we can achieve this for trees. We use a condition which we call the P_4-rule. A decomposition w satisfies the *P_4-rule* if for every induced four-vertex path $P = (e_1, e_2, e_3)$ in G, the following condition holds: if $w(e_2) = \max\{w(e_1), w(e_2), w(e_3)\}$, then $w(e_2) \geq w(e_1) + w(e_3)$. We will prove the following theorem.

Theorem 2. *Let T be a tree. Then $r(T) = \min\{k \mid w$ is a decomposition of size k that satisfies the P_4-rule$\}$.*

Note that for cycles, the equality does not hold. For example, consider the decomposition w of C_6 given by the weights $2, 4, 2, 2, 3, 1$. This decomposition satisfies the P_4 rule but it can be shown using case analysis that there does not exist a labeling ℓ achieving w.

However, we can give a characterization of readability for C_4-free graphs in terms of a parameter that is asymptotically equivalent to readability, using a condition which we call the strict P_4-rule. The strict P_4-rule is identical to the P_4-rule accept that the inequality becomes strict. That is, w satisfies the *strict P_4-rule* if for every induced four-vertex path $P = (e_1, e_2, e_3)$, if $w(e_2) = \max\{w(e_1), w(e_2), w(e_3)\}$, then $w(e_2) > w(e_1) + w(e_3)$. Note that a decomposition that satisfies the strict P_4-rule automatically satisfies the P_4-rule, but not vice-versa. We will prove the following theorem.

Theorem 3. *Let G be a C_4-free bipartite graph. Let $t = \min\{k \mid w$ is a decomposition of size k that satisfies the strict P_4-rule$\}$. Then $t/2 < r(G) \leq t$.*

We note that this characterization cannot be extended to graphs with a C_4. The example in Figure 1a shows a graph with a decomposition which satisfies the strict P_4-rule but it can be shown using case analysis that there does not exists a labeling ℓ achieving this decomposition.

In the remainder of this section, we will prove these two theorems. We first show that an ℓ-decomposition satisfies the P_4-rule (proof in the full version).

Lemma 1. *Let ℓ be an overlap labeling of a bipartite graph G. Then the ℓ-decomposition satisfies the P_4-rule.*

Now, consider a C_4-free bipartite graph $G = (V_s, V_p, E)$ and let w be a decomposition satisfying the P_4-rule. We will prove both Theorem 2 and Theorem 3 by constructing the following labeling. Let us order the edges $e_1, \ldots, e_{|E|}$ in order of non-decreasing weight. For $0 \leq j \leq |E|$, we define the graph $G^j = (V_s, V_p, \{e_i \in E \mid i \leq j\})$. For a vertex u, define $len_j(u) = \max\{w(e_i) \mid i \leq j, e_i$ is incident with $u\}$, if the degree of u in G^j is positive, and 0 otherwise. We will recursively define a labeling ℓ_j of G^j such that $|\ell_j(u)| = len_j(u)$ for all u. The initial labeling ℓ_0 assigns ϵ to every vertex. Suppose we have a labeling ℓ_j for G^j, and $e_{j+1} = (u, v)$. Recall that because w satisfies the P_4-rule and G is C_4-free, $w(u, v) \geq len_j(u) + len_j(v) = |\ell_j(u)| + |\ell_j(v)|$. (Note that the inequality holds also in the case when one of the two summands is 0.) Let A be a (possibly empty) string of length $w(u, v) - |\ell_j(u)| - |\ell_j(v)|$ composed of non-repeating characters that do not exist in ℓ_j. Define ℓ_{j+1} as $\ell_{j+1}(x) = \ell_j(x)$ for all $x \notin \{u, v\}$, and $\ell_{j+1}(u) = \ell_{j+1}(v) = \ell_j(v) \cdot A \cdot \ell_j(u)$. We denote the labeling of G as $\ell = \ell_{|E|}$. We will slightly abuse notation in this section, ignoring the fact that a labeling must have labels of the same length. This is inconsequential, because strings can always be padded from the beginning or end with distinct characters without affecting any overlaps.

First, we state a useful Lemma, that two vertices share a character in the labeling only if they are connected by a path (proof in the full version).

Lemma 2. *Let c be a character that is contained in $\ell_j(u)$ and in $\ell_j(v)$, for some pair of distinct vertices. Then there exists a path between u and v in G^j.*

We are now ready to show that ℓ achieves w for trees, and, if w also satisfies the strict P_4-rule, for C_4-free graphs.

Lemma 3. *Let G be a C_4-free bipartite graph and let w be a decomposition that satisfies the P_4-rule. Then the above defined labeling ℓ achieves w if w satisfies the strict P_4-rule or if G is acyclic.*

Proof sketch. We prove by induction on j that ℓ_j achieves w on G^j. Suppose that the Lemma holds for ℓ_j and consider the effect of adding $e_{j+1} = (u, v)$. Notice that to obtain ℓ_{j+1} we only change labels by adding outer characters, hence, any two vertices that overlap by i in ℓ_j will also overlap by i in ℓ_{j+1}. Moreover, only the labels of u and v are changed, and an overlap between u

and v of length $w(u,v)$ is created. It remains to show that no shorter overlap is created between u and v and that no new overlap is created involving u or v, except the one between u and v.

For the cases when $w(u,v) > |\ell_j(u)|+|\ell_j(v)|$ or $w(u,v) = |\ell_j(v)|$ or $w(u,v) = |\ell_j(u)|$, we show in the full version that ℓ_{j+1} achieves w on G^{j+1}. We similarly show in the full version that the case when $w(u,v) = |\ell_j(u)|+|\ell_j(v)|$ and $\ell_j(u) \neq \epsilon \neq \ell_j(v)$ cannot arise if w satisfies the strict P_4-rule.

Now, assume that G is acyclic, and suppose for the sake of contradiction that the new labeling creates an overlap between v and a vertex $u' \neq u$ (the case of an overlap between u and $v' \neq v$ is symmetric). Consider the character c at position $|\ell_j(v)| + 1$ of $\ell_{j+1}(v)$. The length of the overlap between $\ell_{j+1}(v)$ and $\ell_{j+1}(u') = \ell_j(u')$ must be greater than $|\ell_j(v)|$, otherwise it would have been an overlap in ℓ_j. Thus, $\ell_j(u')$ must contain c. By construction of v's new label, $\ell_j(u)$ must also contain c. Applying Lemma 2, there must be a path between u' and u in G^j. On the other hand, the overlap between v and u' spans $(\ell_j(v))[1]$, and hence $\ell_j(v)$ and $\ell_j(u')$ must share a character. Applying Lemma 2, there must exist a path between u' and v in G^j. Consequently, there exists a path from u to v in G^j. Combining this path with $e_{j+1} = (u,v)$, we get a cycle in G^{j+1}, which is a contradiction.

Finally, suppose for the sake of contradiction that $\ell_{j+1}(u)$ overlaps $\ell_{j+1}(v)$ by some $k < w(u,v)$. By the induction hypothesis, $k > |\ell_j(v)|$. Consider the last character c of $\ell_j(v)$. It must also appear as the inner position $i = k - |\ell_j(v)| + 1$ in $\ell_{j+1}(u)$. Since $k \leq w(u,v) - 1$, we have $i \leq w(u,v) - |\ell_j(v)| = |\ell_j(u)|$, and the i^{th} inner position in $\ell_{j+1}(u)$ is also the the i^{th} inner position in $\ell_j(u)$. Applying Lemma 2 to c in $\ell_j(v)$ and $\ell_j(u)$, there must exist a path between u and v in G^j. Combining this path with $e_{j+1} = (u,v)$, we get a cycle in G^{j+1}, which is a contradiction. □

We can now prove Theorems 2 and 3.

Proof of Theorem 2. Let $t = \min\{k \mid w \text{ is a decomposition of size } k \text{ that satisfies the } P_4\text{-rule}\}$. First, let w be a decomposition of size t satisfying the P_4-rule. Lemma 3 states that the above defined labeling ℓ achieves w and so $r(T) \leq \max_e(w_e) = t$. For the other direction, consider an overlap labeling b of T of minimum length. By Lemma 1, the b-decomposition satisfies the P_4-rule. Hence, $r(T) = len(b) \geq t$. □

Proof of Theorem 3. Let w be a decomposition of size t satisfying the strict P_4-rule. By Lemma 3, the above defined labeling ℓ achieves w and so $r(G) \leq \max_e(w_e) = t$. On the other hand, let b be an overlap labeling of length $r(G)$. Define $w(e) = 2\text{ov}_b(e) - 1$, for all $e \in E(G)$. We claim that w satisfies the strict P_4-rule, which will imply that $t \leq \max_e w(e) = 2r(G) - 1$. To see this, let e_1, e_2, e_3 be the edges of an arbitrary induced P_4. Observe that $w(e_2) = \max\{w(e_1), w(e_2), w(e_3)\}$ if and only if $\text{ov}_b(e_2) = \max\{\text{ov}_b(e_1), \text{ov}_b(e_2), \text{ov}_b(e_3)\}$. Furthermore, it can be algebraicly verified that if $\text{ov}_b(e_2) \geq \text{ov}_b(e_1) + \text{ov}_b(e_3)$ then $w(e_2) > w(e_1) + w(e_3)$. By Lemma 1, the b-decomposition satisfies the P_4-rule and, therefore, w satisfies the strict P_4-rule. □

3.2 General graphs

In the previous subsection, we derived graph theoretic characterizations of readability that are exact for trees and approximate for C_4-free bipartite graphs. Unfortunately, for a general graph, it is not clear how to construct an overlap labeling from a decomposition satisfying the P_4-rule (as we did in Lemma 3). In this subsection, we will consider an alternate rule (HUB-rule), which we then use to construct an overlap labeling.

Given $G = (V_s, V_p, E)$ and a decomposition w of size k, we define G_i^w, for $i \in [k]$, as a graph with the same vertices as G and edges given by $E(G_i^w) = \{e \in E \mid w(e) = i\}$. When w is obvious from the context, we will write G_i instead of G_i^w. Observe that the edge sets of $G_1^w, \ldots, G_k^w$ form a partition of E. We say that w satisfies the *hierarchical-union-of-bicliques rule*, abbreviated as the *HUB-rule*, if the following conditions hold: i) for all $i \in [k]$, G_i^w is a disjoint union of bicliques, and ii) if two distinct vertices u and v are non-isolated twins in G_i^w for some $i \in \{2, \ldots, k\}$ then, for all $j \in [i-1]$, u and v are (possibly isolated) twins in G_j^w. An example of a decomposition satisfying the HUB-rule is any $w : E \to [k]$ such that G_1^w is an (arbitrary) disjoint union of bicliques and $G_2^w, \ldots, G_k^w$ are matchings. We can show that the decomposition implied by any overlap labeling must satisfy the HUB-rule (proof in the full version).

Lemma 4. *Let ℓ be an overlap labeling of a bipartite graph G. Then the ℓ-decomposition satisfies the HUB-rule.*

We define *the HUB number of G* as the minimum size of a decomposition of G that satisfies the HUB-rule, and denote it by $hub(G)$. Observe that a decomposition of a graph into matchings (i.e. each G_i^w is a matching) satisfies the HUB-rule. By König's Line Coloring Theorem, any bipartite graph G can be decomposed into $\Delta(G)$ matchings, where $\Delta(G)$ is the maximum degree of G. Thus, $hub(G) \in [\Delta(G)]$. Clearly, a graph G has $hub(G) = 1$ if and only if G is a disjoint union of bicliques. The HUB number captures readability in the sense that the readability of a graph family is bounded (by a uniform constant independent of the number of vertices) if and only if its HUB number is bounded. This is captured by the following theorem:

Theorem 4. *Let G be a bipartite graph. Then $hub(G) \leq r(G) \leq 2^{hub(G)} - 1$.*

In the remainder of this section, we will prove this theorem. The first inequality directly follows from Lemma 4 because, by definition of readability, there exists an overlap labeling ℓ of length $r(G)$. Then the ℓ-decomposition of G is of size $r(G)$ and satisfies the HUB-rule, implying $hub(G) \leq r(G)$. To prove the second inequality, we will need to show:

Lemma 5. *Let w be a decomposition of size k satisfying the HUB-rule of a bipartite graph G. Then there is an overlap labeling of G of length $2^k - 1$.*

The second inequality of Theorem 4 follows directly by choosing a minimum decomposition satisfying the HUB-rule, in which case $k = hub(G)$. Thus, it only remains to prove Lemma 5.

We now define the labeling t that is used to prove Lemma 5. Our construction of the labeling applies the following operation due to Braga and Meidanis [BM02]. Given two vertices $u \in V_s$ and $v \in V_p$, a labeling t, and a filler character a not used by t, the *BM operation* transforms t by relabeling both u and v with $t(v) \cdot a \cdot t(u)$.

We start by labeling G_1 as follows: each biclique B in G_1 gets assigned a unique character a_B, and each node v in a biclique B gets label $t(v) = a_B$. Next, for $i \in [k-1]$, we iteratively construct a labeling of $G_1 \cup \cdots \cup G_{i+1}$ from a labeling t of $G_1 \cup \cdots \cup G_i$. We show by induction that the constructed labeling has an additional property that all twins in $G_1 \cup \cdots \cup G_{i+1}$ have the same labels and that the length of the labeling is $2^{i+1} - 1$. Observe that the labeling of G_1 satisfies this property.

We choose a unique (not previously used) character a_B for each biclique B of G_{i+1}. If B consists of a single vertex v, then we assign to v the label $a_B \cdot t(v)$ if $v \in V_s$, and $t(v) \cdot a_B$ if $v \in V_p$. Otherwise, since w satisfied the HUB-rule, all vertices in $B \cap V_s$ are twins in $G_1 \cup \cdots \cup G_i$ and, by the induction hypothesis, are assigned the same labels in t. Analogously, t will assign the same labels to all nodes in $B \cap V_p$. Consider an arbitrary edge (u, v) in B. We apply the BM operation with character a_B to (u, v) and assign the resulting label $t(v) \cdot a_B \cdot t(u)$ to all nodes in B. This completes the construction of labeling of $G_1 \cup \cdots \cup G_{i+1}$. Observe that it assigns the same labels to all twins in $G_1 \cup \cdots \cup G_{i+1}$, and that the length is $2^{i+1} - 1$. To complete the proof of Theorem 4, we show in the full version that the final labeling is an overlap labeling of G.

Note that if w is a decomposition into matchings, then our labeling algorithm behaves identically to the Braga-Meidanis (BM) algorithm [BM02]. However, in the case that w is of size $o(\Delta(G))$, our labeling algorithm gives a better bound than BM. For example, for the $n \times n$ biclique, our algorithm gives a labeling of length 1, while BM gives a labeling of length $2^n - 1$.

4 Lower and upper bounds on readability

In this section, we prove several lower and upper bounds on readability, making use of the characterizations of the previous section.

4.1 Almost all graphs have readability $\Omega(n/\log n)$

In this subsection, we show that, in both the bipartite and directed graph models, there exist graphs with readability at least $\Omega(n/\log n)$, and that in fact almost all graphs have at least this readability.

Theorem 5. *Almost all graphs in $\mathcal{B}_{n\times n}$ (and, respectively, $\mathcal{D}_n$) have readability $\Omega(n/\log n)$. When restricted to a constant sized alphabet, almost all graphs in $\mathcal{B}_{n\times n}$ (and, respectively, $\mathcal{D}_n$) have readability $\Omega(n)$.*

Proof (constant sized alphabet case). We prove the lemma by a counting argument. Since there are n^2 pairs of nodes in $[n]^2$ that can form edges in a graph in

$\mathcal{B}_{n\times n}$, the size of $\mathcal{B}_{n\times n}$ is 2^{n^2}. Let a be the size of the alphabet. The number of labelings of $2n$ nodes with strings of length s is at most a^{2ns}. In particular, labelings of length $s = n/(3\log a)$ can generate no more than $a^{2n^2/(3\log a)} = 2^{2n^2/3}$ bipartite graphs, which is in $o(2^{n^2})$. Consequently, almost all graphs in $\mathcal{B}_{n\times n}$ have readability $\Omega(s) = \Omega(n/\log a) = \Omega(n)$. The proof for $\mathcal{D}_n$ is analogous and is omitted. The proof for variable sized alphabets is given in the full version. □

4.2 Distinctness and a graph family with readability $\Omega(n)$

In this subsection, we will give a technique for proving lower bounds and use it to show a family of graphs with readability $\Omega(n)$. For any two vertices u and v, the *distinctness* of u and v is defined as $DT(u,v) = \max\{|N(u) \setminus N(v)|, |N(v) \setminus N(u)|\}$. The *distinctness* of a bipartite graph G, denoted by $DT(G)$, is defined as the minimum distinctness of any pair of vertices that belong to the same part of the bipartition. The following lemma relates the distinctness and the readability of graphs that are not matchings (for a matching, the readability is 1, provided that it has at least one edge, and 0 otherwise).

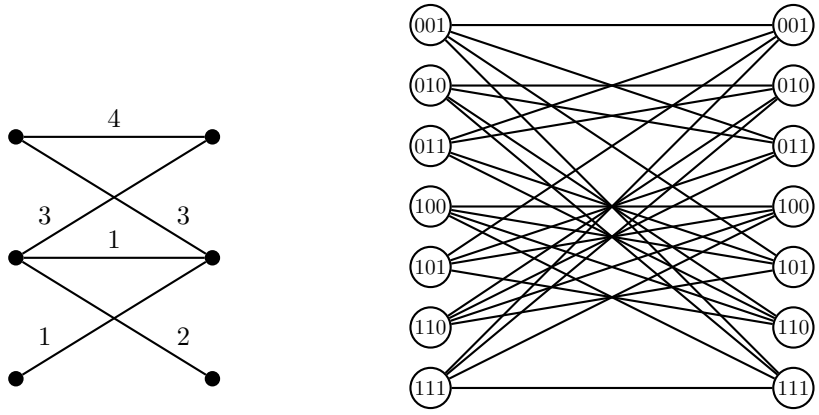
Lemma 6. *For each bipartite graph G that is not a matching, $r(G) \geq DT(G) + 1$.*

Proof. By Theorem 4, it suffices to show that $DT(G) \leq hub(G) - 1$. Let $h = hub(G)$, let $w : E(G) \to [h]$ be a minimum decomposition of G satisfying the HUB-rule, and consider the graphs $G_i = G_i^w$, for $i \in [h]$. We need to show that $DT(G) \leq h - 1$. Suppose first that each G_i is a matching. Then, since w is a decomposition of G, we have $\Delta(G) \leq h$. Moreover, since G is not a matching, it has a pair of distinct vertices, say u and v, with a common neighbor, which implies $DT(G) \leq DT(u,v) \leq \Delta(G) - 1 \leq h - 1$.

Suppose now that there exists an index $j \in [h]$ such that G_j is not a matching, and let j be the maximum such index. Then, there exist two distinct vertices in G, say u and v, that have a common neighbor in G_j, and therefore belong to the same biclique of G_j. It follows that u and v are non-isolated twins in G_j. Since w is satisfies the HUB-rule, this implies that u and v are twins in each G_i with $i \in [j-1]$. Consequently, for each vertex x in G adjacent to u but not to v, the unique G_i with $(u,x) \in E(G_i)$ satisfies $i > j$. By the choice of j, each such G_i is a matching, and hence there can be at most $h - j$ such vertices x. Thus $|N(u) \setminus N(v)| \leq h - j$ and similarly $|N(v) \setminus N(u)| \leq h - j$, which implies the desired inequality $DT(G) \leq DT(u,v) \leq h - j \leq h - 1$. □

While the distinctness is a much simpler graph parameter than the HUB number, simplicity comes with a price. Namely, the distinctness does not share the nice feature of the HUB number, that of being bounded on exactly the same sets of graphs as the readability. In Section 4.3, we show the existence of graphs (specifically, trees) of distinctness 1 and of arbitrary large readability.

We now introduce a family of graphs, inspired by the Hadamard error correcting code, and apply Lemma 6 to show that their readability is at least linear



(a) (b)

Fig. 1: (a) Illustration that Theorem 3 cannot be extended to graphs with a C_4. Example of a graph and decomposition that satisfies the strict P_4-rule, yet no overlap labeling ℓ exists that achieves it. (b) The graph H_3. The strings on the vertices correspond to the k-bit codeword vectors.

in the number of nodes. We define H_k as the bipartite graph with vertex sets $V_s = \{v_s \mid v \in \{0,1\}^k \setminus \{0^k\}\}$ and $V_p = \{v_p \mid v \in \{0,1\}^k \setminus \{0^k\}\}$ and edge set

$$E(H_k) = \left\{(v_s, v_p) \in V_s \times V_p \;\mid\; \sum_{i=1}^{k} v_s[i]v_p[i] \equiv 1 \pmod 2\right\}.$$

In other words, each vertex has a non-zero k-bit codeword vector associated with it and two vertices are adjacent if the inner product of their codewords is odd. Let $n = 2^k$. Graph H_k has $2(n-1)$ vertices, all of degree $n/2$, and thus $(n-1)n/2$ edges. Figure 1b illustrates H_3.

In the full version, we show that every pair of vertices in the same part of the bipartition of H_k has exactly $n/4$ common neighbors. This implies that the distinctness of H_k is $n/4$. Combining this with Lemma 6, we obtain the following theorem.

Theorem 6. *$r(H_k) \geq n/4+1$.*

This lower bound also translates to directed graphs: applying Theorem 1, there exists digraphs of readability $\Omega(n)$. A major open question is: Do there exist graphs that have exponential readability? We conjecture that they do, and that the graph family H_k has exponential readability. However, since distinctness is $O(n)$, we note that Lemma 6 is insufficient for proving stronger than $\Omega(n)$ lower bounds on the readability.

4.3 Trees

The purely graph theoretic characterization of readability given by Theorem 2 allows us to derive a sharp upper bound on the readability of trees. Recall that

the *eccentricity* of a vertex u in a connected graph G is defined as $ecc_G(u) = \max_{v \in V(G)} dist_G(u, v)$, where $dist_G(u, v)$ is the number of edges in a shortest path from u to v. The *radius* of a graph G is defined as the minimum eccentricity of a vertex in G, that is $radius(G) = \min_{u \in V(G)} \max_{v \in V(G)} dist_G(u, v)$.

Theorem 7. *For every tree T, $r(T) \leq radius(T)$, and this bound is sharp. More precisely, for every $k \geq 0$, there exists a tree T such that $r(T) = radius(T) = k$.*

Proof. Let T be a tree. If $T = K_1$ (the one-vertex tree), then $radius(T) = r(T) = 0$ (note that assigning the empty string to the unique vertex of v results in an overlap labeling of T). Now, let T be of radius $r \geq 1$ and let $v \in V(T)$ be a vertex of T of minimum eccentricity (that is, $ecc_T(v) = r$). Consider the distance levels of T from v, that is, $V_i = \{w \in V(T) \mid dist_T(v, w) = i\}$ for $i \in \{0, 1, \ldots, r\}$. Also, for all $i \in [r]$, let E_i be the set of edges in T connecting a vertex in V_{i-1} with a vertex in V_i. Then $\{E_1, \ldots, E_r\}$ is a partition of $E(T)$ and the decomposition $w : E(T) \to [r]$ given by $w(e) = i$ if and only if $e \in E_i$ is well defined. We claim that w satisfies the P_4-rule. Let $P = (v_1, v_2, v_3, v_4)$ be an induced P_4 in T, and let $i = w(v_1, v_2)$, $j = w(v_2, v_3)$, $k = w(v_3, v_4)$. Suppose that $j = \max\{i, j, k\}$. We may assume without loss of generality that $v_2 \in V_{j-1}$ and $v_3 \in V_j$. Since T is a tree, v_2 is the only neighbor of v_3 in V_{j-1}, which implies that $v_4 \in V_{j+1}$ and consequently $k = j + 1$, contrary to the assumption $j = \max\{i, j, k\}$. Thus, the P_4-rule is trivially satisfied for w. By Theorem 2, we have $r(T) \leq \max_{e \in E(T)} w(e) = r = radius(T)$.

To show that for every $k \geq 0$ there exists a tree T with $r(T) = radius(T) = k$, we proceed by induction. We will construct a sequence $\{(T_i, v_i)\}_{i \geq 0}$ where T_i is a tree, v_i is a vertex in T_i with $ecc_{T_i}(v_i) \leq i$, the degree of v_i in T_i is i, and $r(T_i) = radius(T_i) = i$. For $i = 0$, take $(T_0, v_0) = (K_1, v_0)$ where v_0 is the unique vertex of K_1. This clearly has the desired properties. For $i \geq 1$, take i disjoint copies of (T_{i-1}, v_{i-1}), say (T_{i-1}^j, v_{i-1}^j) for $j \in [i]$, add a new vertex v_i, and join v_i by an edge to each v_{i-1}^j for $j \in [i]$. Let T_i be the so constructed tree. Clearly, the degree of v_i in T_i is i, and $ecc_{T_i}(v_i) \leq 1 + ecc_{T_i}(v_{i-1}) \leq 1 + (i - 1) = i$, which implies that $radius(T_i) \leq i$. On the other hand, we will show that $r(T_i) \geq i$, which together with inequality $r(T_i) \leq radius(T_i)$ will imply the desired conclusion $radius(T_i) = r(T_i) = i$. Suppose for a contradiction that $r(T_i) < i$. Then, by Lemma 1, there exists a decomposition w of T_i of size $i - 1$ satisfying the P_4-rule. In particular, this implies $i \geq 2$. Since the degree of v_i in T_i is i, there exist two edges incident with v_i, say (v_i, v_{i-1}^j) and (v_i, v_{i-1}^k) for some $j \neq k$ such that $w(v_i, v_{i-1}^j) = w(v_i, v_{i-1}^k)$. Let w_1 denote this common value. Let x be a neighbor of v_{i-1}^j in T_{i-1}^j. (Note that x exists since v_{i-1}^j is of degree $i - 1 \geq 1$ in T_{i-1}^j.) Then, $(x, v_{i-1}^j, v_i, v_{i-1}^k)$ is an induced P_4 in T_i. We claim that $w(x, v_{i-1}^j) > w_1$. Indeed, if $w(x, v_{i-1}^j) \leq w_1$ then we have $\max\{w(x, v_{i-1}^j), w(v_{i-1}^j, v_i), w(v_i, v_{i-1}^k)\} = \max\{w(x, v_{i-1}^j), w_1, w_1\} = w_1$, while $w_1 \not\geq w_1 + w(x, v_{i-1}^j)$, contrary to the P_4-rule. Since x was an arbitrary neighbor of v_{i-1}^j in T_{i-1}^j, we infer that every edge e in T_{i-1}^j incident with v_{i-1}^j satisfies $w(e) > w_1$. In particular, this leaves a set of at most $i - 2$ different

values that can appear on these $i-1$ edges (the value w_1 is excluded), and hence again there must be two edges of the same weight, say w_2. Clearly, $w_2 > w_1$ and $i > 2$. Proceeding inductively, we construct a sequence of edges $e_1, e_2, \ldots, e_i$ forming a path in T_i from v_i to a leaf and satisfying $w_1 < w_2 < \ldots < w_i$, where $w_i = w(e_i)$. This implies that all the weights $w_1, \ldots, w_i$ are distinct, contrary to the fact that the range of w is contained in the set $[i-1]$. This contradiction shows that $r(T_i) \geq i$ and completes the proof. □

Note that for every $k \geq 2$, the tree T_k of radius k constructed in the proof of Theorem 2 has a pair of leaves in the same part of the bipartition and is therefore of distinctness 1. This shows that the readability of a graph cannot be upper-bounded by any function of its distinctness (cf. Lemma 6).

5 Conclusion

In this paper, we define a graph parameter called readability, and initiate a study of its asymptotic behavior. We give purely graph theoretic parameters (i.e., without reference to strings) that are exactly (respectively, asymptotically) equivalent to readability of trees (respectively, C_4-free graphs. However, for general graphs, the HUB number is equivalent to readability only in the sense that it is bounded on the same set of graphs. While an ℓ-decomposition always satisfies the HUB-rule, the converse is not true. For example, a decomposition of P_4 with weights $4, 5, 3$ satisfies the HUB-rule but cannot be achieved by an overlap labeling (by Lemma 1). For this reason, the upper bound given by Lemma 5 leaves a gap with the lower bound of Lemma 4. We are able to describe other properties that an ℓ-decomposition must satisfy (not included in the paper), however, we are not able to exploit them to close the gap. It is a very interesting direction to find other necessary rules that would lead to a graph theoretic parameter that would more tightly match readability on general graphs than the HUB number.

Consider $r(n) = \max\{r(D) \mid D \text{ is a digraph on } n \text{ vertices}\}$. We have shown $r(n) = \Omega(n)$ and know from [BM02] that $r(n) = O(2^n)$. Can this gap be closed? Do there exist graphs with readability $\Theta(2^n)$ (as we conjecture), or, for example, is readability always bounded by a polynomial in n? Questions regarding complexity are also unexplored, e.g., given a digraph, is it NP-hard to compute its readability? For applications to bioinformatics, the length of reads can be said to be poly-logarithmic in the number of vertices. It would thus be interesting to further study the structure of graphs that have poly-logarithmic readability.

Acknowledgements. P.M. and M.M. would like to thank Marcin Kamiński for preliminary discussions. P.M. was supported in part by NSF awards DBI-1356529 and CAREER award IIS-1453527. M.M. was supported in part by the Slovenian Research Agency (I0-0035, research program P1-0285 and research projects N1-0032, J1-5433, J1-6720, and J1-6743). S.R. was supported in part by NSF CAREER award CCF-0845701, NSF award AF-1422975 and the Hariri Institute for Computing and Computational Science and Engineering at Boston University.

References

[BBT13] Guy Bresler, Ma'ayan Bresler, and David Tse. Optimal assembly for high throughput shotgun sequencing. *BMC Bioinformatics*, 14(Suppl 5):S18, 2013.

[BFK+02] Jacek Błażewicz, Piotr Formanowicz, Marta Kasprzak, Petra Schuurman, and Gerhard J. Woeginger. DNA sequencing, Eulerian graphs, and the exact perfect matching problem. In *Graph-Theoretic Concepts in Computer Science*, pages 13–24. Springer, 2002.

[BFKK02] Jacek Błażewicz, Piotr Formanowicz, Marta Kasprzak, and Daniel Kobler. On the recognition of de Bruijn graphs and their induced subgraphs. *Discrete Mathematics*, 245(1):81–92, 2002.

[BHKdW99] Jacek Blazewicz, Alain Hertz, Daniel Kobler, and Dominique de Werra. On some properties of DNA graphs. *Discrete Applied Mathematics*, 98(1):1–19, 1999.

[BM02] Marília D. V. Braga and Joao Meidanis. An algorithm that builds a set of strings given its overlap graph. In *LATIN 2002: Theoretical Informatics, 5th Latin American Symposium, Cancun, Mexico, April 3-6, 2002, Proceedings*, pages 52–63, 2002.

[BM08] John A. Bondy and Uppaluri S. R. Murty. *Graph Theory*, volume 244 of *Graduate Texts in Mathematics*. Springer, New York, 2008.

[GP14] Theodoros P. Gevezes and Leonidas S. Pitsoulis. Recognition of overlap graphs. *Journal of Combinatorial Optimization*, 28(1):25–37, 2014.

[IW95] Ramana M. Idury and Michael S. Waterman. A new algorithm for DNA sequence assembly. *Journal of Computational Biology*, 2(2):291–306, 1995.

[LZ07] Xianyue Li and Heping Zhang. Characterizations for some types of DNA graphs. *Journal of Mathematical Chemistry*, 42(1):65–79, 2007.

[LZ10] Xianyue Li and Heping Zhang. Embedding on alphabet overlap digraphs. *Journal of Mathematical Chemistry*, 47(1):62–71, 2010.

[MGMB07] Paul Medvedev, Konstantinos Georgiou, Gene Myers, and Michael Brudno. Computability of models for sequence assembly. In *Algorithms in Bioinformatics*, pages 289–301. Springer, 2007.

[MKS10] Jason R Miller, Sergey Koren, and Granger Sutton. Assembly algorithms for next-generation sequencing data. *Genomics*, 95(6):315–327, 2010.

[Mye05] Eugene W. Myers. The fragment assembly string graph. In *ECCB/JBI*, page 85, 2005.

[NP09] Niranjan Nagarajan and Mihai Pop. Parametric complexity of sequence assembly: theory and applications to next generation sequencing. *Journal of computational biology*, 16(7):897–908, 2009.

[NP13] Niranjan Nagarajan and Mihai Pop. Sequence assembly demystified. *Nature Reviews Genetics*, 14(3):157–167, 2013.

[PSW03] Rudi Pendavingh, Petra Schuurman, and Gerhard J. Woeginger. Recognizing DNA graphs is difficult. *Discrete Applied Mathematics*, 127(1):85–94, 2003.

[Swe00] Z Sweedyk. A $2\frac{1}{2}$-approximation algorithm for Shortest Superstring. *SIAM Journal on Computing*, 29(3):954–986, 2000.

[TU88] Jorma Tarhio and Esko Ukkonen. A greedy approximation algorithm for constructing shortest common superstrings. *Theoretical Computer Science*, 57(1):131–145, 1988.