Compression of Dynamic Graphs Generated by a Duplication Model

Krzysztof Turowski, Abram Magner, and Wojciech Szpankowski
Center for Science of Information,
Department of Computer Science, Purdue University, West Lafayette, IN, USA
Email: krzysztof.szymon.turowski@gmail.com, abram10@gmail.com, spa@cs.purdue.edu

***Abstract*—We continue building up the information theory of non-sequential data structures such as trees, sets, and graphs. In this paper, we consider dynamic graphs generated by a full duplication model in which a new vertex selects an existing vertex and copies all of its neighbors. We ask how many bits are needed to describe the labeled and unlabeled versions of such graphs. We first estimate entropies of both versions and then present asymptotically optimal compression algorithms up to a constant term. Interestingly, for the full duplication model the labeled version needs $\Theta(n)$ bits while its unlabeled version (structure) can be described by $\Theta(\log n)$ bits due to a significant amount of symmetry (i.e., the cardinality of the automorphism group of graphs generated by this model is on average quite high).**

I. Introduction

Complex systems can often be modeled as dynamic graphs. In these systems, patterns of interactions evolve in time, determining emergent properties, associated function, robustness, and security of the system. There are several broad questions whose answers shed light on the evolution of such dynamic networks: (i) how many bits are required to best describe such a network and its structure (i.e., unlabeled underlying graph); (ii) how to infer underlying dynamic processes governing network evolution; (iii) how to infer information about previous states of the network; and (iv) how to predict the forward evolution of the network state. In this conference paper we deal with the first question (i.e., labeled and unlabeled graph compression).

To better understand the evolution of network structural properties, several probabilistic models have been proposed, including, e.g., the preferential attachment, duplication-divergence, Cooper-Frieze, and fit-get richer models [2], [6], [10], [24].

Clearly, some models are more suitable to certain types of data than others. For example, it has been claimed that the preferential attachment mechanism [2] plays a strong role in the formation of citation networks [23]. However, due to the high power law exponent of their degree sequence (greater than 2) and lack of community structure [6], they are not likely to describe well biological networks such as protein interaction networks or gene regulatory networks [19]. For such networks another model, known as the vertex-copying model, or simply the *duplication model*, has been claimed as a better fit. In the vertex-copying model, one picks an existing vertex and inserts its clone, possibly with some random modifications, depending on the exact variation of the model [6], [20], [13]. Experimental results show that these variations on the duplication model better capture salient features of protein interaction networks [22] than does the preferential attachment model.

In this paper we present comprehensive information-theoretic results for the full duplication model in which every new vertex is a copy of some older vertex. We establish precisely (that is, within a $O(1)$ additive error) the entropy for both unlabeled and labeled graphs generated by this model and design asymptotically optimal compression algorithms that match the entropies up to a constant term. Interestingly, we shall see that the entropy of labeled graphs is $H(G_n) = \Theta(n)$, while the structural entropy (the entropy of the isomorphism class of a random graph from the model, denoted by $S(G_n)$) is significantly smaller: $H(S(G_n)) = \Theta(\log n)$. Thus, the vast majority of information of the labeled graphs in this model is present in the labeling itself, not in the underlying graph structure. In contrast, the entropy of the labeled and unlabeled graphs generated by, e.g., the preferential attachment model is $\Theta(n \log n)$ [17].

Clearly, given its simplicity, this model should be regarded as a stepping stone toward a better understanding of more advanced models of this type. The extensions are typically defined by a fixed-probability mix of the full duplication model and other rules, such as no-duplication or uniform attachment. We shall deal with such models in a forthcoming paper.

Graph compression has enjoyed a surge in popularity in recent years, as the recent survey [3] shows. However, rigorous information-theoretic results are still lacking, with a few notable exceptions. The rigorous information-theoretic

This work was supported by NSF Center for Science of Information (CSoI) Grant CCF-0939370, and in addition by NSF Grant CCF-1524312, NIH Grant 1U01CA198941-01, and NSC Grant UMO-2016/21/B/ST6/03146.

analysis of graph compression (particularly in the unlabeled case) was initiated by Choi and Szpankowski [5], who analyzed structural compression of Erdős-Rényi graphs (see also [1]). The authors of [5] presented a compression algorithm that provably achieves asymptotically the first two terms of the structural entropy. In Magner et al. [17] the authors precisely analyzed the labeled and structural entropies and gave asymptotically optimal compression algorithms for preferential attachment graphs. There has been recent work on universal compression schemes, including in a distributed scenario, by Delgosha and Anantharam [8], [9]. Additionally, several works deal with compression of trees [11], [12], [18], [25].

The full duplication model was almost exclusively analyzed in the context of the typical properties such as degree distribution [6]. It was shown that the average degree depends strongly on the initial conditions[16]. It was also proved that the asymptotic degree distribution fails to converge, yet it exhibits power-law behavior with exponent dependent on the lowest nonzero degree in the initial graph [21]. Other parameters studied in the context of duplication models are number of small cliques [14] or degree-degree correlation [4]. To the best of our knowledge the entropy and compression of duplication models was not discussed previously in any available literature.

II. Full duplication model

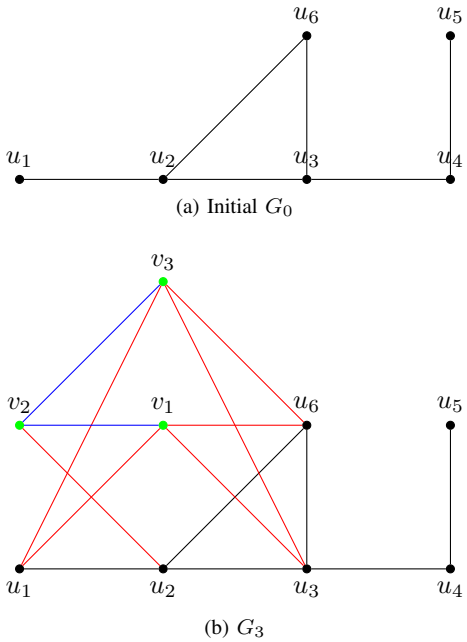
In this section we define the full duplication model and present some of its properties.

A. Definitions

The full duplication model is defined as follows: let us denote by G_0 a given graph on n_0 vertices for some fixed constant n_0. Then, for any $1 \leq i \leq n$ we obtain G_i from G_{i-1} by choosing one of the vertices of G_{i-1} (denoted by v) uniformly at random, attaching to the graph a new vertex v_i and adding edges between v_i and all vertices adjacent to v. Note that v and v_i are not connected – although if one wants to achieve higher clustering, the results in this paper can be straightforwardly applied to the model in which we add not only edges between v_i and the neighbors of v, but also between v_i and v. Observe that G_n has $n+n_0$ vertices. Also, properties of G_n heavily depend on G_0 and its structure, which we assume to be fixed.

Throughout this paper, we will refer to the vertices of G_0 as $U = \{u_1, \ldots, u_{n_0}\}$ and to all other vertices from G_n as $V = \{v_1, \ldots, v_n\}$. We denote by $N_n(v)$ the neighborhood of the vertex v, that is, all vertices that are adjacent to v in G_n. Sometimes we drop the subscript, if the size of the graph is clear from the context.

An example of the duplication process is presented in Figure 1. On the top, we show the original G_0 on 6 vertices, and on the bottom we plot G_3 with new vertices such that v_1 is a copy of u_2, v_2 is a copy of u_1, and v_3 is a copy of v_1.



(a) Initial G_0

(b) G_3

Fig. 1: Example graph growth in the full duplication model

Here, due to the limited space, we restrict our analysis to asymmetric G_0 (i.e., the underlying automorphism group is of size 1); however, extensions to general G_0 are rather straightforward. We observe that typically even moderate-sized graphs are likely to be asymmetric.

B. Basic Properties

Let us introduce the concept of a parent and an ancestor of a vertex. We say that w is the *parent* of v (denoted by $w = P(v)$), when v was copied from w at some time $1 \leq i \leq n$. We say that $w \in U$ is the *ancestor* of v (denoted by $w = A(v)$), when there exist vertices $v_{i_1}, \ldots, v_{i_k}$ such that $w = P(v_{i_1})$, $v_{i_j} = P(v_{i_{j+1}})$ for $1 \leq j \leq k-1$, and $v_{i_k} = v$. For convenience we write that if $u \in U$, then $P(u) = u$ and $A(u) = \{u\}$. Note that the ancestor of any given vertex is unique. In our example from Figure 1 u_2 is an ancestor of both v_1 and v_3, but only a parent of v_1 and not v_3.

The neighborhood of a vertex is closely tied to its ancestor, as the following lemma shows:

Lemma 1. *We have $A(v_i) = A(v_j)$ if and only if $N(v_i) = N(v_j)$.*

Proof: If $u = A(v_i)$, then $N(v_i) = N(u)$ in G_j, $j \geq i$. We can prove this by induction on the number of intermediate vertices whose ancestor was u before v_i. For the base case, when this number is 0, v_i is the first vertex to duplicate u, and the claim is trivially true. Now, if there are $k \geq 1$

vertices whose ancestor is u, and all of their neighborhoods are equal to that of u, an additional vertex choosing to duplicate any vertex included in their shared neighborhood is added to all of their neighborhoods. Additionally, a vertex that chooses to duplicate one of the descendants of u inherits its neighborhood from all of them.

Now, we prove the other direction: i.e., if $N(v_i) = N(u)$, then $A(v_i) = u$. Since G_0 is simple and asymmetric, $N(u) \neq N(u')$ in G_0 for any $u' \neq u$. This implies that $N(v_i) \neq N(u')$, which in turn implies, by the discussion above, that $A(v_i) \neq u'$ for any $u' \neq u$. Thus, $A(v_i) = u$, as desired. ∎

For $1 \leq i \leq n_0$ let $C_{i,n} = |\{w \in G_n \colon A(w) = u_i\}$, that is, the number of vertices from G_n that are ultimately copies of u_i (including u_i itself).

The sequence of variables $(C_{i,n})_{i=1}^{n_0}$ can be described as a ball and urn model with n_0 urns. At time $n = 0$ each urn contains exactly one ball. Each iteration consists of picking an urn at random, proportionally to the number of balls in each bin – that is, with probability $\frac{C_{i,n}}{\sum_{j=1}^{n_0} C_{j,n}}$ – and adding a new ball to this urn. It is known [15] that the joint distribution of $(C_{i,n})_{i=1}^{n_0}$ is the *Dirichlet multinomial distribution* denoted as $Dir(n, \alpha_1, \ldots, \alpha_{n_0})$, with $K = n_0$ and $\alpha_i = 1$ for $1 \leq i \leq n_0$:

$$\mathbb{P}\left((C_{i,n})_{i=1}^{n_0} = (k_i + 1)_{i=1}^{n_0}\right) = \frac{\Gamma(n+1)\Gamma(n_0)}{\Gamma(n+n_0)}.$$

Each variable $C_{i,n}$ is identically distributed – though not independent, as we know that $\sum_{i=1}^{n_0} C_{i,n} = n$ – so we may analyze the properties of $C_n \sim C_{i,n}$ for every $1 \leq i \leq n_0$. It has the *beta-binomial distribution* $BBin(n, \alpha, \beta)$ with parameters $\alpha = 1$, $\beta = n_0 - 1$. That is:

$$\begin{aligned}\mathbb{P}(C_n = k+1) &= \binom{n}{k} \frac{B(k+1, n+n_0-k-1)}{B(1, n_0-1)} \\ &= (n_0-1)\binom{n}{k} B(k+1, n+n_0-k-1)\end{aligned}$$

for any $k \geq 0$, where $B(x,y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)}$ is the Euler beta function.

Therefore, since $C_n \sim BBin(n, 1, n_0 - 1) + 1$ we know immediately that $\mathbb{E}C_n = \frac{n}{n_0} + 1$.

We will also need some other properties discussed below:

Lemma 2. *If $X \sim BBin(n, \alpha, \beta)$, then $\mathbb{E}[\log(X+1)] = \log n + O(1)$.*

Lemma 3. *If $X \sim BBin(n, \alpha, \beta)$, then*

$$\begin{aligned}\mathbb{E}[(X+1)\log(X+1)] &= n \log n \frac{\alpha}{\alpha+\beta} \\ &+ n \frac{\alpha\left(\psi(\alpha+1) - \psi(\alpha+\beta+1)\right)\log e}{\alpha+\beta} + \log n + O(1).\end{aligned}$$

where $\psi(x) = \frac{\Gamma'(x)}{\Gamma(x)}$ is the Euler digamma function.

For all integers r, s we have $\psi(r) - \psi(s) = H_{r-1} - H_{s-1}$ and $H_s = H_{s-1} + \frac{1}{s}$ (where H_j denotes the jth harmonic number). Therefore,

Corollary 1. *Asymptotically,*

$$\begin{aligned}\mathbb{E}[\log C_n] &= \log n + O(1) \\ \mathbb{E}[C_n \log C_n] &= n \log n \frac{1}{n_0} + n \frac{(1 - H_{n_0})\log e}{n_0} \\ &\quad + \log n + O(1).\end{aligned}$$

III. Main theoretical results

As discussed in the introduction, our goal is to present results for the duplication graphs on structural parameters which are fundamental to statistical and information-theoretic problems involving the information shared between the labels and the structure of a random graph. In graph structure compression the goal is to remove label information to produce a compact description of a graph structure.

Formally, the labeled graph compression problem can be phrased as follows: one is given a probability distribution $\mathcal{G}_n$ on graphs on n vertices, and the task is to exhibit a pair of mappings (i.e., a source code) (E, D), where E maps graphs to binary strings satisfying the standard prefix code condition, and D maps binary strings back to graphs, such that, for all graphs H, $D(E(H)) = H$, and the *expected code length* $\mathbb{E}[|E(G)|]$, with $G \sim \mathcal{G}_n$, is minimized. The standard source coding theorem tells us that the fundamental limit for this quantity is $H(G)$.

The *unlabeled* version of this problem relaxes the invertibility constraint on the encoder and decoder. In particular, we only require $D(E(H)) \cong H$; i.e., the decoder only outputs a graph isomorphic to H. Again, the optimization objective is to minimize the expected code length. Thus, in effect, the source code efficiently describes the isomorphism type of its input. Denoting by $S(G)$ the isomorphism type of G, the fundamental limit for the expected code length is the *structural entropy* of the model, which is given by $H(S(G))$.

There is a relation between the labeled entropy $H(G)$ and structural entropy $H(S(G))$. To express it succinctly for a broad class of graph models we need the automorphism group* $\mathrm{Aut}(G)$, the positive-probability *labeled representatives* of a given structure, and the set $\Gamma(G)$ of *feasible permutations* of G; i.e., the set of permutations of G that yield a graph that has positive probability under the random graph model in question. See [5], [17] for more details.

Now, we are ready to present a relation between $H(G)$ and $H(S(G))$. The following lemma was proved in [17].

Lemma 4. *We have, for any graph model G_n in which all positive-probability labeled graphs that are isomorphic have the same probability,*

$$H(G_n) - H(S(G_n)) = \mathbb{E}[\log|\Gamma(G_n)|] - \mathbb{E}[\log|\mathrm{Aut}(G_n)|].$$

*An automorphism of a graph is a permutation that preserves edge relations. In other words, it is a permutation which, when applied to the graph, yields the same graph (note that, in mathematical literature, a graph is by default labeled).

In Appendix A we prove the following results regarding the expected logarithms of the sizes of the automorphism group and feasible permutation set for samples G_n from the full duplication model.

Lemma 5. *We have*

$$\mathbb{E}[\log|\operatorname{Aut}(G_n)|] = n\log n - nH_{n_0}\log e + \frac{3n_0}{2}\log n + O(1)$$

for large n.

In Appendix B we prove the following.

Lemma 6. *Asymptotically*

$$\mathbb{E}[\log|\Gamma(G_n)|] = n\log n - n\log e + \frac{2n_0+1}{2}\log n + O(1).$$

We now estimate the structural entropy. Recalling that we assume throughout that the initial graph G_0 is asymmetric, it may be seen that the isomorphism type of G_n is entirely specified by the vector $(C_{i,n})_{i=1}^{n_0}$. We know that $(C_{i,n})_{i=1}^{n_0}$ has the Dirichlet multinomial distribution with $\alpha_i = 1$ for $1 \le i \le n_0$. Therefore

$$\mathbb{P}((C_{i,n})_{i=1}^{n_0} = (k_i+1)_{i=1}^{n_0}) = nB(n,n_0)\prod_{i=1}^{n_0} k_i^{-1}B(k_i,1)^{-1}$$

leading to

$$\begin{aligned}
&H(S(G_n)) = H(C_{i,n}) + O(1)\\
&= -\sum_{(c_i)_{i=1}^{n_0}} \mathbb{P}((C_{i,n})_{i=1}^{n_0} = (c_i)_{i=1}^{n_0})\\
&\quad \log\mathbb{P}((C_{i,n})_{i=1}^{n_0} = (c_i)_{i=1}^{n_0})\\
&= -\sum_{(c_i)_{i=1}^{n_0}} nB(n,n_0)\log(nB(n,n_0))\\
&= -\log n - \log B(n,n_0)\\
&= -\log n - \left(n+\frac{1}{2}\right)\log n\\
&\quad + \left(n+n_0+\frac{1}{2}\right)\log(n+n_0) + O(1)\\
&= -\log n + n_0\log n + O(1) = (n_0-1)\log n + O(1).
\end{aligned}$$

The last two lines follow respectively from the Stirling approximation and the Taylor expansion of $\log(n+n_0)$.

This leads to our first main result:

Theorem 1. *For large n we have*

$$H(S(G_n)) = (n_0-1)\log n + O(1).$$

To compute the graph entropy $H(G)$ we can use Lemma 4 together with Theorem 1; however, it is instructive to compute it with "bare hands" which is presented in Appendix C.

Theorem 2. *For large n*

$$H(G_n) = n(H_{n_0}-1)\log e + \frac{n_0-1}{2}\log n + O(1).$$

IV. Algorithmic results

In this section we present asymptotically optimal algorithms for compression of labeled and unlabeled graphs generated according to the full duplication model.

A. Retrieval of parameters from G_n

In order to present efficient compression algorithms for the duplication model, we must first reconstruct G_0 from G_n and find values of n_0 and n. This is relatively easy to accomplish. Indeed, for a given labeled G_n let $(w_1, w_2, \ldots, w_{n+n_0})$ be its vertices in the order in which they appeared (in particular, $(w_1,\ldots,w_{n_0}) = (u_1,\ldots,u_{n_0})$ and $(w_{n_0+1},\ldots,w_{n_0+n}) = (v_1,\ldots,v_n)$). It is sufficient to find the smallest k such that $N(w_k) = N(w_i)$ for some $1 \le i < k$. Then $n_0 = k-1$ and G_0 is induced by the sequence $(w_1,\ldots,w_{k-1})$.

The case for unlabeled graphs is similar: we know (for details see Lemma 7 in the Appendix B) that the sequence of the first n_0 vertices of the graph (that is, G_0) contains exactly one vertex from each set $\{v\colon A(v) = u_i\}$.

From Lemma 1 we know that $A(v) = A(w)$ iff $N(v) = N(w)$, so it is sufficient to scan all vertices of G_n and split them into sets such that v and w belongs to the same set iff $N(v) = N(w)$. Then, we pick one vertex from each set to from G_0. Obviously, n_0 and n may be extracted from the sizes of G_0 and G_n.

For example, for a graph in Fig. 1(b) we may identify the following classes of vertices with identical neighborhoods: $\{u_1, v_2\}$, $\{u_2, v_1, v_3\}$, $\{u_3\}$, $\{u_4\}$ and $\{u_5\}$. Therefore, we know that $n_0 = 6$, $n = 3$ and the G_0 is isomorphic to a graph induced, for example, by the set $\{v_2, v_3, u_3, u_4, u_5\}$.

In summary, we establish the following result.

Theorem 3. *For a given labeled G_n or its unlabeled version $S(G_n)$, we can retrieve its n, n_0 and G_0 (in the case of structure up to isomorphisms of G_0) in polynomial time in terms of n.*

B. Unlabeled graphs

A trivial algorithm CompressUnlabeledSimple for unlabeled compression writes down a sequence $(C_{i,n})_{i=1}^{n_0}$ associated with our G_n as $\log n$-bit numbers. This always requires $n_0\log n$ bits, so $\mathbb{E}L_{SU}(n) = n_0\log n + O(1)$, where L_{SU} denotes the code length of our proposed scheme. By Theorem 1 this achieves the fundamental limit to within a multiplicative factor of $1 + \frac{1}{n_0-1}$.

However, it is easy to design an optimal algorithm up $O(1)$ term. The pseudocode of an optimal algorithm, called CompressUnlabeledOpt, based on arithmetic coding, is as follows:

function CompressUnlabeledOpt($S(G_n)$, $S(G_0)$)
Fix a labeled representative G_0 of $S(G_0)$.

Fix any ordering $(v_1, \dots, v_n)$ of the vertices of $S(G_n) \setminus S(G_0)$
$a \leftarrow 0$, $b \leftarrow 1$
for $i = 1, 2, \dots, n_0$ **do**
  $C[i] = 1$
  for $j = 1, 2, \dots, n$ **do**
    if $N(v_j) = N(u_i)$ **then**
      $C[i] = C[i] + 1$

for $i = n_0, n_0 - 1, \dots, 1$ **do**
  $start \leftarrow \frac{\Gamma(n+1)\Gamma(i)}{\Gamma(n+i)} \sum_{j=1}^{k-1} \frac{\Gamma(n+n_0-j)}{\Gamma(n+n_0-1)}$
  $end \leftarrow start + \frac{\Gamma(n+1)\Gamma(k)\Gamma(n+n_0-k)}{\Gamma(n+k)\Gamma(n+n_0-j)}$
  $b \leftarrow a + (b-a) \cdot start$
  $a \leftarrow a + (b-a) \cdot end$
  $n \leftarrow n - C[i] + 1$
$p \leftarrow b - a$, $x \leftarrow \dfrac{a+b}{2}$
return $G_0 \| [\text{first } \lceil -\log p \rceil + 1 \text{ bits of } x]$

The next finding proves that COMPRESSUNLABELEDOPT is asymptotically optimal.

Theorem 4. *Algorithm* COMPRESSUNLABELEDOPT *is optimal up to a constant term for unlabeled graphs compression, when the graph is generated by the full duplication model.*

Proof: It is sufficient to observe that

$$\begin{aligned}
&\mathbb{P}((C_{i,n})_{i=1}^{n_0} = (k_i+1)_{i=1}^{n_0}) = \\
&= \mathbb{P}((C_{i,n})_{i=1}^{n_0} = (k_i+1)_{i=1}^{n_0} | C_{n_0,n} = k_{n_0}+1) \\
&\quad \cdot \mathbb{P}(C_{n_0,n} = k_{n_0}+1) \\
&= \mathbb{P}\left((C_{i,n})_{i=1}^{n_0-1} = (k_i+1)_{i=1}^{n_0-1} \Big| \sum_{i=0}^{n_0-1} C_{i,n} = k_{n_0}+1 \right) \\
&\quad \cdot \mathbb{P}(C_{n_0,n} = k_{n_0}+1) \\
&= \mathbb{P}((C_{i,n-k_{n_0}})_{i=1}^{n_0-1} = (k_i+1)_{i=1}^{n_0-1}) \\
&\quad \cdot f\left(k_i \Big| \sum_{j \le i} k_j, 1, n_0 - 1 \right)
\end{aligned}$$

where f is the probability mass function of the beta-binomial distribution.

It is known that the marginal distribution of the Dirichlet multinomial distribution is the beta-binomial distribution. And if we fix a value of one coordinate to c for the Dirichlet multinomial distribution, it is known that the resulting distribution is also Dirichlet multinomial, but with $n_0 - 1$ coordinates and all values summing up to $n - c$.

We repeat this process – but note that the whole process stops when we have Dirichlet multinomial distribution distribution on with 2 coordinates:

$$\begin{aligned}
&\mathbb{P}((C_{i,n})_{i=1}^{2} = (k_i+1)_{i=1}^{2}) \\
&\quad = \mathbb{P}(C_{1,n} = k_1 + 1 | C_{1,n} = k_1 + 1) \mathbb{P}(C_{2,n} = k_1 + 1) \\
&\quad = f(k_1 | k_1 + k_2, 1, 1).
\end{aligned}$$

Putting all together we get

$$\begin{aligned}
&\mathbb{P}((C_{i,n})_{i=1}^{n_0} = (k_i+1)_{i=1}^{n_0}) \\
&= \prod_{i=2}^{n_0} f(c_i | \sum_{j \le i} k_j, 1, n_0 - i) \\
&= \prod_{i=2}^{n_0} (n_0 - 1) \frac{\Gamma(\sum_{j \le i} k_j + 1)\Gamma(\sum_{j<i} k_j + n_0 - i)}{\Gamma(\sum_{j \le i} k_j + n_0 - i + 1)\Gamma(\sum_{j<i} k_j + 1)} \\
&= (n_0 - 1)! \frac{\Gamma(n+1)}{\Gamma(n + n_0 - 1)} = nB(n, n_0)
\end{aligned}$$

This matches the entropy, so by the properties of arithmetic coding (see [7]), we know that $\mathbb{E}L_O(S(G_n)) \le H((C_{i,n})_{i=0}^{n_0}) + 2 + \mathbb{E}L_O(S(G_0)) + \mathbb{E}L_O(\pi) = H(S(G_n)) + O(1)$ (where L_O denotes the code length). This completes the proof. ∎

C. Labeled graphs

We note tat the labeled graph G_n is equivalent to a sequence $(A(v_i))_{i=1}^{n}$ for a given (labeled) G_0 (which obviously can be encoded using a constant number of bits).

A trivial algorithm COMPRESSLABELEDSIMPLE just writes all $A(v_i)$ as $\log n_0$-bit numbers. Clearly, this always gives us a codeword with length $n \log n_0$ and therefore (as we need to encode G_0) $\mathbb{E}L_{SL}(n) = n \log n_0 + O(1)$ (where L_{SL} denotes the code length). From Theorem 2 it is known that this algorithm is asymptotically $(1 + \frac{1-\gamma}{\log n_0})$-approximate, where γ is Euler-Mascheroni constant.

It is easy to design an asymptotically optimal algorithm up to the $O(1)$ term. Indeed, the sequence of $A(v_i)$ is random with $\mathbb{P}(A(v_i) = u_j) = \frac{C_{j,i-1}}{i+n_0-1}$ for $1 \le i \le n$, $1 \le j \le n_0$. Therefore, given G_{i-1} we know the conditional probabilities of G_i and we may construct another algorithm based on arithmetic coding.

The pseudocode of the optimal algorithm is as follows:

function COMPRESSLABELEDOPT(G_n, G_0)
  $a \leftarrow 0$, $b \leftarrow 1$
  for $i = 1, 2, \dots, n_0$ **do**
    $C[i] = 1$
  for $i = 1, 2, \dots, n$ **do**
    for $j = 1, 2, \dots, n_0$ **do**
      if $N(v_i) = N(u_j)$ **then**
        $start \leftarrow \sum_{k=1}^{j-1} \frac{C[k]}{n_0+i-1}$
        $end \leftarrow start + \frac{C[j]}{n_0+i-1}$
        $b \leftarrow a + (b-a) \cdot start$
        $a \leftarrow a + (b-a) \cdot end$
        $C[i] = C[i] + 1$
  $p \leftarrow b - a$, $x \leftarrow \dfrac{a+b}{2}$
  return $G_0 \| [\text{first } \lceil -\log p \rceil + 1 \text{ bits of } x]$

The next theorem proves that COMPRESSLABELEDOPT is asymptotically optimal.

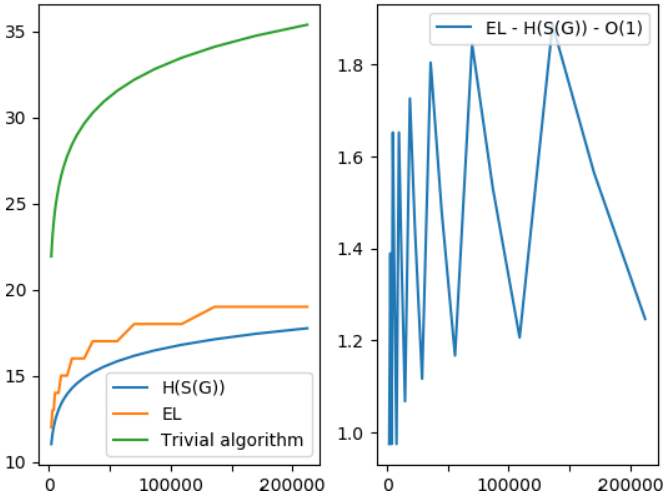
Theorem 5. *Algorithm* CompressLabeledOpt *is optimal up to a constant term for labeled graph compression, when the graph is generated by the full duplication model.*

Proof: By the properties of arithmetic encoding (see [7]), we know that $\mathbb{E}L_O(G_n) \leq H(G_n|G_0)+2+\mathbb{E}L_O(G_0) = H(G_n) + O(1)$, where L_O denotes the code length. ■

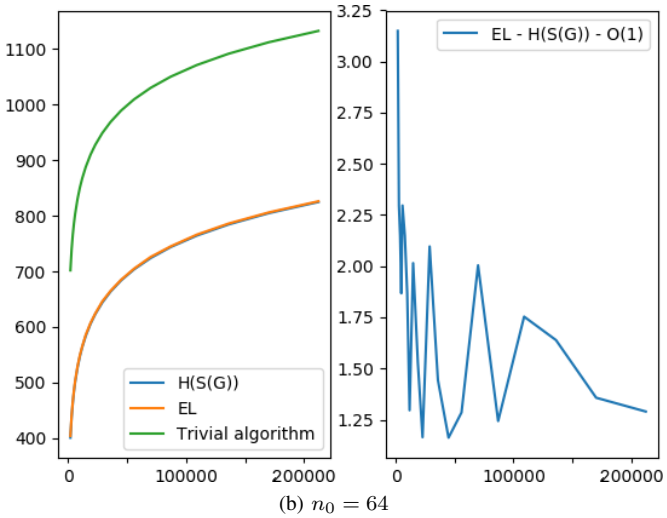
Note that these two algorithms differ only in that the optimal one updates the probabilities at each step and the second fixes them to a constant value of $1/n_0$.

V. Experimental results

We implemented our compression algorithms for both labeled and unlabeled graphs and ran them on synthetic data, generated according to the full duplication model. Figure 2 reports the results of compression for unlabeled graphs, while Figure 3 for labeled graphs.

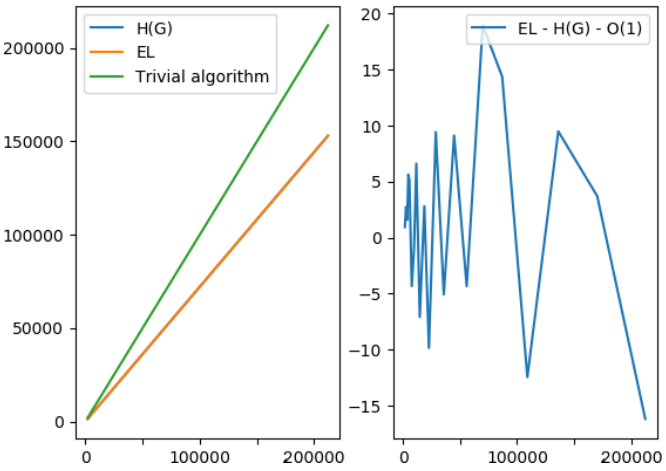


(a) $n_0 = 2$

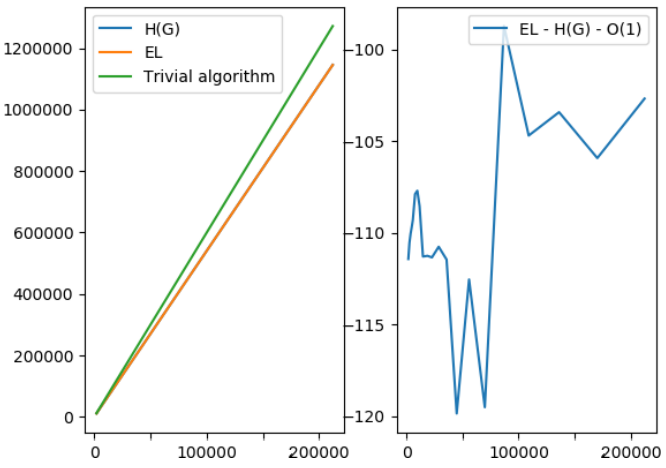


(b) $n_0 = 64$

Fig. 2: Entropy and redundancy for unlabeled graphs. The x axis is n, and the y axis is a number of bits.



(a) $n_0 = 2$



(b) $n_0 = 64$

Fig. 3: Entropy and redundancy for labeled graphs. The x axis is n, and the y axis is a number of bits.

For unlabeled graphs the experimental error does converge quickly to values less than 2, which is consistent with the well-known properties of arithmetic encoding. For labeled graphs we observe divergence in error – which is apparently very small compared to the size of graphs. Unfortunately, the size of the sample space is very large compared to the number of sampled graphs (here we generated $2 \cdot 10^6$ graphs for every n considered); therefore, the average length of compressed graphs varies significantly between experiments.

References

[1] Emmanuel Abbe. Graph compression: the effect of clusters. In *Proc. of the fifty-fourth annual Allerton conference*, 2016.

[2] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Rev. Mod. Phys*, 74:47–97, 2002.

[3] M. Besta and T. Hoefler. Survey and taxonomy of lossless graph compression and space-efficient graph representations. *Preprint*, 2018.

[4] Stefano Boccaletti, Dong-Uk Hwang, and Vito Latora. Growing hierarchical scale-free networks by means of nonhierarchical processes. *I. J. Bifurcation and Chaos*, 17:2447–2452, 2007.

[5] Yongwook Choi and W. Szpankowski. Compression of graphical structures: fundamental limits, algorithms, and experiments. *IEEE Trans. Inf. Theor.*, 58(2):620–638, February 2012.

[6] Fan Chung, Linyuan Lu, T. Gregory Dewey, and David J. Galas. Duplication models for biological networks. *Journal of Computational Biology*, 10(5):677–687, 2003. PMID: 14633392.

[7] T. Cover and J. Thomas. *Elements of Information Theory*. John Wiley & sons, 2nd edition, 2006.

[8] Payam Delgosha and Venkat Anantharam. Universal lossless compression of graphical data. In *Proc. of the International Symposium on Information Theory (to appear)*, 2017.

[9] Payam Delgosha and Venkat Anantharam. Distributed compression of graphical data. In *Proc. of the International Symposium on Information Theory (to appear)*, 2018.

[10] Alan Frieze and Michał Karoński. *Introduction to Random Graphs*. Cambridge University Press, 2016.

[11] Z. Gołębiewski, A. Magner, and W. Szpankowski. Entropy of some general plane trees. In *2017 IEEE International Symposium on Information Theory (ISIT)*, pages 301–305, June 2017.

[12] D. Hucke and M. Lohrey. Universal tree source coding using grammar-based compression. In *2017 IEEE International Symposium on Information Theory (ISIT)*, pages 1753–1757, June 2017.

[13] I. Ispolatov, P. L. Krapivsky, and A. Yuryev. Duplication-divergence model of protein interaction network. *Phys. Rev. E*, 71:061911, Jun 2005.

[14] Iaroslav Ispolatov, Pavel Krapivsky, I Mazo, and A Yuryev. Cliques and duplication-divergence network growth. *New Journal of Physics*, 7:145, 07 2005.

[15] N. Johnson, A. Kemp, and S. Kotz. *Univariate Discrete Distributions*. Wiley, 2005.

[16] J. Kim, P. L. Krapivsky, B. Kahng, and S. Redner. Infinite-order percolation and giant fluctuations in a protein interaction network. *Phys. Rev. E*, 66:055101, Nov 2002.

[17] Tomasz Łuczak, Abram Magner, and Wojciech Szpankowski. Structural information and compression of scale-free graphs. preprint, 2017.

[18] A. Magner, K. Turowski, and W. Szpankowski. Lossless compression of binary trees with correlated vertex names. *Trans. Information Theory*, 64, 2018.

[19] Mark Newman. *Networks: An Introduction*. Oxford University Press, 2010.

[20] Romualdo Pastor-Satorras, Eric Smith, and Ricard V. Solé. Evolving protein interaction networks through gene duplication. *Journal of Theoretical Biology*, 222(2):199 – 210, 2003.

[21] Alpan Raval. Some asymptotic properties of duplication graphs. *Phys. Rev. E*, 68:066119, Dec 2003.

[22] Mingyu Shao, Yi Yang, Jihong Guan, and Shuigeng Zhou. Choosing appropriate models for protein–protein interaction networks: a comparison study. *Briefings in Bioinformatics*, 15(5):823–838, 2014.

[23] Price Derek De Solla. A general theory of bibliometric and other cumulative advantage processes. *Journal of the American Society for Information Science*, 27(5):292–306, 1976.

[24] Remco van der Hofstad. *Random Graphs and Complex Networks: Volume 1*. Cambridge University Press, 2016.

[25] J. Zhang, E. H. Yang, and J. C. Kieffer. A universal grammar-based code for lossless compression of binary trees. *IEEE Transactions on Information Theory*, 60(3):1373–1386, March 2014.

APPENDIX

A. Proof of Lemma 5

Under the assumption that $|\operatorname{Aut}(G_0)| = 1$ we have $\mathbb{E}[\log|\operatorname{Aut}(G_n)|] = \mathbb{E}\left[\log\prod_{i=1}^{n_0} C_{i,n}!\right]$. To prove it, it is sufficient to notice that all vertices v, w such that $A(v) = A(w)$ can be mapped on one another freely (since they have equal neighborhoods) – but if $A(v) \neq A(w)$, there does not exist any automorphism σ for which v and w are in the same orbit. Precisely, this is because, if such a σ did exist, then one may show that it induces an automorphism of G_0.

$$\begin{aligned}\mathbb{E}[\log|\operatorname{Aut}(G_n)|] &= \mathbb{E}\left[\log\prod_{i=1}^{n_0} C_{i,n}!\right]\\ &= \sum_{i=1}^{n_0}\mathbb{E}[\log C_{i,n}!] = n_0\mathbb{E}[\log C_n!].\end{aligned}$$

We use Stirling's approximation together with Corollary 1 to obtain

$$\begin{aligned}&\mathbb{E}[\log C_n!] =\\ &= \mathbb{E}[C_n\log C_n] - \mathbb{E}C_n\log e + \frac{1}{2}\mathbb{E}[\log C_n] + O(1)\\ &= \mathbb{E}[C_n\log C_n] - n\frac{\log e}{n_0} + \frac{1}{2}\mathbb{E}[\log C_n] + O(1)\\ &= n\log n\frac{1}{n_0} + n\frac{(1-H_{n_0})\log e}{n_0} + \log n\\ &\quad - n\frac{\log e}{n_0} + \log n\frac{1}{2} + O(1).\end{aligned}$$

Finally,

$$\mathbb{E}[\log|\operatorname{Aut}(G_n)|] = n\log n - nH_{n_0}\log e + \frac{3n_0}{2}\log n + O(1).$$

The proof is completed.

B. Proof of Lemma 6

Observe that G_n has $n+n_0$ vertices; therefore, the trivial upper bound is $\Gamma(G_n) \leq (n+n_0)!$. We can do the exact computation of $\Gamma(G_n)$ using the following lemma:

Lemma 7. *Let π be a relabeling of G_n such that it produces a positive-probability graph under the full duplication model. Furthermore, let $(w_1,\ldots,w_{n_0+n}) = (u_1,\ldots,u_{n_0},v_1,\ldots,v_n)$. If $1 \leq \pi(w_i) < \pi(w_j) \leq n_0$, then $A(w_i) \neq A(w_j)$ for any $i \neq j$.*

Proof: Let $(w'_1, w'_2, \ldots w'_{n+n_0})$ be the vector $(\pi^{-1}(w_1),\ldots,\pi^{-1}(w_{n_0+n}))$. Let us define, for each $i \in [n_0] = \{1,\ldots,n_0\}$, $F(i) = \{j\colon A(w'_j) = u_i, \pi(w_j) \in [n_0]\}$; i.e., $F(i)$ is the set of vertices whose ancestor in G_n is u_i and that map under π to the initial n_0 vertices in $\pi(G_n)$. To prove the lemma, we will show that $|F(i)| = 1$.

If $|F(i)| \geq 2$ for any i, then two vertices, $w_h \neq w_k$ are mapped by π to the initial n_0 vertices, and they have identical neighborhoods. This contradicts the assumption that $|\operatorname{Aut}(G_0)| = 1$.

If $|F(i)| = 0$ for any i, then there must exist some $i' \in [n_0]$ with $|F(i)| \geq 2$, which is contradicted by the previous case.

Thus, $|F(i)| = 1$ as claimed for all $1 \leq i \leq n_0$. ∎

Clearly, any permutation of the first n_0 vertices is admissible, since every order of vertices in G_0 is admissible. Moreover, any permutation of last n vertices is admissible, because

we may assume that all chosen vertices for replication are only these already existing in G_0. This, together with the fact that $C_{i,n} = |F(i)|$, leads us to the formula

$$|\Gamma(G_n)| = n_0! n! \prod_{i=1}^{n_0} \binom{C_{i,n}}{1} = n_0! n! \prod_{i=1}^{n_0} C_{i,n}.$$

Then

$$\begin{aligned}\mathbb{E}[\log|\Gamma(G_n)|] &= \log n_0! + \log n! + \sum_{i=1}^{n_0} \mathbb{E}[\log C_{i,n}] \\ &= \log n_0! + \log n! + n_0 \mathbb{E}[\log C_n] \\ &= \log n! + n_0 \log n + O(1),\end{aligned}$$

and the final result $\mathbb{E}[\log|\Gamma(G_n)|] = n\log n - n\log e + (n_0 + \frac{1}{2})\log n + O(1)$ follows from the Stirling approximation.

C. Alternative Proof of Theorem 2

Now we give a direct proof of Theorem 2. First, let us note (for the fixed, asymmetric G_0) the one-to-one relationship between G_n and a sequence $(A(v_1), \ldots, A(v_n))$. Therefore, the entropy of G_n is equal to the entropy of a sequence $(A(v_1), \ldots, A(v_n))$ generated according to the rules.

This is in fact a joint distribution of n sequential steps of ball-and-urn process with n_0 urns, each containing one ball at the beginning. Every chosen ball is removed, its color is reported as i-th value and two new balls of the same color are added instead. The number of balls in urns at a given time is also known as the Dirichlet multinomial distribution with $K = n_0$ and $\alpha_i = 1$ for $1 \le i \le n_0$.

We can compute the total entropy of this distribution the following way. First, start with chain rule:

$$H(G_n|G_0) = \sum_{i=1}^{n} H(G_i|G_{i-1})$$

Then, for every $1 \le i \le n$:

$$\begin{aligned}H(G_i|G_{i-1}) &= \sum_G p_{i-1}(G) H(G_i|G_{i-1} = G) \\ &= \sum_G p_{i-1}(G) \sum_{G'} p_i(G'|G) \log \frac{1}{p_i(G'|G)}\end{aligned}$$

where $p_i(G) = \mathbb{P}(G_i = G)$, $p_i(G'|G) = \mathbb{P}(G_i = G'|G_{i-1} = G)$.

Now observe that $p_i(G'|G) = \mathbb{P}(A(v_i) = u_j) = \frac{C_{j,i-1}(G)}{i+n_0-1}$ if G' is equal to G with attached copy of u_j for some $1 \le j \le n_0$. If G' and G do not meet this condition, then clearly the conditional probability is equal to 0. This lead us to:

$$\begin{aligned}H(G_i|G_{i-1}) &= \sum_G p_{i-1}(G) \sum_{j=1}^{n_0} \frac{C_{j,i-1}(G)}{i+n_0-1} \log \frac{i+n_0-1}{C_{j,i-1}(G)} \\ &= \sum_{j=1}^{n_0} \mathbb{E}\left[\frac{C_{j,i-1}}{i+n_0-1} \log \frac{i+n_0-1}{C_{j,i-1}}\right] \\ &= n_0 \mathbb{E}\left[\frac{C_{i-1}}{i+n_0-1} \log \frac{i+n_0-1}{C_{i-1}}\right] \\ &= \log(i+n_0-1) - \frac{n_0}{i+n_0-1} \mathbb{E}[C_{i-1} \log C_{i-1}].\end{aligned}$$

Here, we need to use a stronger version of Corollary 1

$$\begin{aligned}\mathbb{E}[C_n \log C_n] = \frac{1}{n_0} n \log n + n \frac{(1 - H_{n_0})\log e}{n_0} \\ + \log n + \left(\frac{3}{2} + \frac{1}{2n_0} - H_{n_0}\right)\log e + O(n^{-1})\end{aligned}$$

and slightly rearrange the terms:

$$\begin{aligned}\mathbb{E}[C_{i-1} \log C_{i-1}] = \log(i-1)\frac{i+n_0-1}{n_0} \\ + (1 - H_{n_0}) \log e \frac{i+n_0-1}{n_0} \\ + \left(\frac{1}{2} + \frac{1}{2n_0}\right)\log e + O(i^{-1}).\end{aligned}$$

Note that this formula holds only for $i > 1$, but from the definition it is straightforward that $\mathbb{E}[C_0 \log C_0] = 0$.

Putting this all together we get

$$\begin{aligned}H(G_n) &= \sum_{i=1}^{n_0} (\log(n+i-1) - \log i) \\ &\quad - (n-1)(1 - H_{n_0})\log e \\ &\quad - \frac{n_0+1}{2}(H_{n+n_0-1} - H_{n_0-1})\log e + \sum_{i=1}^{n} O(i^{-1}) \\ &= n_0 \log n - (1 - H_{n_0})(n-1)\log e \\ &\quad - \frac{n_0+1}{2} \ln n \log e + \sum_{i=1}^{n} O(i^{-1}) \\ &= (H_{n_0} - 1)n + \frac{n_0 - 1}{2}\log n + O(\log n)\end{aligned}$$

but, as we see from the theorem, the actual error term is even smaller ($O(1)$ instead of $O(\log n)$). This completes the proof.