Compression of Vertex Transitive Graphs

Bruce Litow, [*] Narsingh Deo, [†] and Aurel Cami [†]

Abstract

We consider the lossless compression of vertex transitive graphs. An undirected graph $G = (V, E)$ is called vertex transitive if for every pair of vertices $x, y \in V$, there is an automorphism σ of G, such that $\sigma(x) = y$. A result due to Sabidussi, guarantees that for every vertex transitive graph G there exists a graph mG (m is a positive integer) which is a Cayley graph. We propose as the compressed form of G a finite presentation (X, R) , where (X, R) presents the group Γ corresponding to such a Cayley graph mG. On a conjecture, we demonstrate that for a large subfamily of vertex transitive graphs, the original graph G can be completely reconstructed from its compressed representation.

1 Introduction

The complex networks that describe systems in nature and society are typically very large, often with hundreds of thousands of vertices. Examples of such networks include the World Wide Web, the Internet, semantic networks, social networks, energy transportation networks, the global economy *etc.,* (see *e.g.*, [16]). Given a graph that represents such a large network, an important problem is its lossless compression, *i.e.*, obtaining a smaller size representation of the graph, such that the original graph can be fully restored from its compressed form. We note that, in general, graphs are incompressible, *i.e.*, the vast majority of graphs with n vertices require $\Omega(n^2)$ bits in any representation. This can be seen by a simple counting argument. There are $2^{n\cdot(n-1)/2}$ labelled, undirected graphs, and at least $2^{n\cdot(n-1)/2}/n!$ unlabelled, undirected graphs. The denominator $n!$ overestimates the number of isomorphs per labelled graph. Taking the logarithm, we see that on average at least $n \cdot (n-1)/2 - n \log n$ bits are needed to represent a graph. Despite this observation, there are families of graphs that

[*] School of Information Technology, James Cook University, Townsville, Qld. 4811, Australia, `bruce@cs.jcu.edu.au`

[†] School of Computer Science, University of Central Florida, Orlando, FL 32816, USA, {`deo, acami`}`@cs.ucf.edu`

can be compressed significantly. For example, the compression of trees is discussed in [4], [14], [2]; of graphs of bounded genus in [6], [11]; and of planar graphs and maps in [10], [5], [8]. More recently, the compression of the web graphs (graphs that model the World Wide Web) has received considerable attention, *e.g.,* see [1], [15], [3], [18].

In this paper, we investigate the compression of vertex transitive graphs. A graph G is said to be vertex transitive if for every pair of vertices x, y there exists an automorphism σ of G such that $\sigma(x) = y$. A precise definition of vertex transitivity in terms of adjacency matrices is given in Section 2. Vertex transitivity is a natural condition required of interconnection networks (*e.g.,* see [9]), for computing or data communications because the network 'looks the same' from any vertex. In this study, we first want to explore how much compression is possible for vertex transitive graphs, without focusing on the computational efficiency of the compression algorithm.

Our starting point in investigating the compressibility of vertex transitive graphs is the observation that a finite group always has finite presentations. We give an elementary proof of this in the appendix. Below we outline the main steps of our scheme, and the rest of the paper develops the details.

1. Given a vertex transitive graph G, compute a positive integer m such that mG is a Cayley graph (a result due to Sabidussi [17]). The adjacency matrix of the graph mG is $O_m \otimes M_G$, where O_m is the $m \times m$ matrix all of whose entries are 1, $\otimes$ is the Kronecker product, and M_G is the adjacency matrix of G.

2. Compute a 'small' finite presentation, (X, R) for the group Γ associated with mG.

3. Use the pair $m, (X, R)$ as a compressed representation of G.

4. Reconstruct a graph G' isomorphic to mG from (X, R). This requires the result that the word problem for a finite presentation of a group known to be finite is decidable.

5. Using m, solve the Sabidussi 'division' problem, *i.e.*, extract a graph G'' isomorphic to G from G'. The main result of this paper is that this problem is well-posed for a large family of graphs G. Whether it is always well-posed is an open problem.

Referring to Steps 1 and 2 of our scheme, we expect that m and (X, R) can be computed such that $\log m$ plus the size in bits of (X, R) will be very much smaller than the number of vertices in G.

As mentioned, our objective is an exploration of the extent to which vertex transitive graphs are compressible. While computational efficiency is not our focus, we show that all of the steps, except Step 5 (which requires the Conjecture 1) are computable.

1.1 Groups and presentations

We review some results from group theory. For more details see, *e.g.*, [12]. Our approach is closer to formal language theory than [12], but is equivalent to the material there. All groups in this paper are finite, although some of our results can also be applied to infinite groups. Γ denotes a group, $g \cdot g'$ denotes the group product of g and g', g^{-1} is the inverse of g, and ι is the group identity element. A set X of elements of a group Γ is said to generate Γ if every element of Γ can be expressed as a finite product of elements of X. For example, a cyclic group can be generated by a single element. The Klein group of functions $x, -x, 1/x, -1/x$ under composition can not be generated by any single one of its elements (however, it can be generated by the two elements $\{-x, 1/x\}$). As another example, the symmetric group S_n of permutations on numbers $\{1, 2, \ldots, n\}$ can be generated by the set of transpositions on $\{1, 2, \ldots, n\}$. If X generates Γ, it is convenient (though not necessary), to require that for any $g \in X$, $g^{-1} \in X$ too. We remark in passing that this requirement is not superfluous for an infinite group some of whose elements do not have finite order.

A finitely generated presentation (X, R) consists of a finite set of symbols X called abstract generators and $R \subseteq X^+$ called relators. In analogy with generators of a group, X is a set of the form: $\{x_1, \ldots, x_a, x_1^{-1}, \ldots, x_a^{-1}\}$, and R always contains $x_i x_i^{-1}$ and $x_i^{-1} x_i$ for $i = 1, \ldots, a$. If R is also finite, the presentation is said to be finite. The next theorem shows that every finitely generated presentation corresponds to a group.

Theorem 1 *Every finitely generated presentation defines a unique group up to isomorphism.*

Proof : Let (X, R) be a finitely generated presentation. The set of relators, R, induces a congruence $\simeq_R$ on X^* as follows (recall that a congruence $\simeq$ on X^* is a an equivalence relation such that $w \simeq w'$ and $v \simeq v'$ imply that $wv \simeq w'v'$). Define a symmetric binary relation $\sim_R$ on X^* by $w \sim_R v$ if $w = w'rw''$ and $v = w'w''$, where $r \in R$. Note that for any $r \in R$, $r \sim_R \lambda$, where λ is the empty string. Define $\simeq_R$ to be the reflexive, transitive closure of $\sim_R$. It is straightforward to show that, indeed, $\simeq_R$ is a congruence.

Furthermore, it can be shown that the set of equivalence classes $X^*/\simeq_R$ forms a group under concatenation. Indeed, if $[w]$ denotes the $\simeq_R$-class containing w, then $[w][v] = [wv]$, which implies that $X^*/\simeq_R$ is closed under concatenation. The equivalence class $[\lambda]$ represents the identity element. Finally, for any element $w = p_1 p_2 ... p_k \in X^+$, the inverse of $[w]$ is the class $[w^{-1}]$ where $w^{-1} = {p_k}^{-1} {p_{k-1}}^{-1} ... {p_1}^{-1}$ and for $i = 1, ..., k$, ${p_i}^{-1}$ is equal to x_i or ${x_i}^{-1}$ if p_i is equal to ${x_i}^{-1}$ or x_i, respectively. □

Any group isomorphic to $X^*/\simeq_R$ is said to be presented by (X, R). Note that the set $P = \{[p] : p \in X\}$ generates $X^*/\simeq_R$. Furthermore if Γ is presented by (X, R) and $\tau : X^*/\simeq_R \to \Gamma$ is an isomorphism, then the set $\tau(P)$ generates Γ. As an example, consider the presentation (X, R) with $X = \{a, b\}$ and $R = \{a^3, b^2, aba^{-1}b^{-1}\}$. It can be shown (see [12]) that

$$X^*/\simeq_R = \{[\lambda], [a], [a^2], [b], [ab], [a^2 b]\},$$

i.e., any group presented by (X, R) has order six.

The next theorem, whose proof is given in the Appendix, states a result which is, in some sense, the converse of Theorem 1.

Theorem 2 *Every finite group has a finite presentation.*

We turn now to the discussion of the word problem for a presentation (X, R) which is defined as follows: given $w, v \in X^*$, decide whether $w \simeq_R v$. The word problem is undecidable even for finite presentations. However, if a finite presentation is known to correspond to a finite group, the situation changes. It is interesting to note that an upper bound on the order of the group is not needed. A sketch of the main idea of the following proof is given in Problem 8 on page 30 of [12].

Theorem 3 *If $X^*/\simeq_R$ is known to be finite for a finite presentation (X, R), then the associated word problem is decidable.*

Proof : We begin by showing that the set of words $w \simeq_R \lambda$ can be computably listed. We describe the listing algorithm in stages.
Stage 0**:** List the set $\{\lambda\} \cup R$.
Stage $(k+1)$**:** For each b listed at Stage k, list all words $b'zb''$, where $z \in R$ and $b = b'b''$. Also, list all strings cc' where $b = czc'$ and $z \in R$.

The algorithm clearly lists only $w \simeq_R \lambda$, and every such element will be listed by the definition of $\simeq_R$.

We turn to the word problem. Given $w, v \in X^*$, we want to determine whether $wv^{-1} \simeq_R \lambda$. Let f be a computable bijection from ordered pairs of nonnegative integers to the nonnegative integers. For example,

$$f(a,b) = \frac{(a+b)^2 + a + 3b}{2} \ ,$$

the Cantor mapping, will do. See [13]. The algorithm proceeds in stages, indexed by the nonnegative integers. It is not intended to be efficient.

Stage $f(i,j)$: If $j = 0$, go to Stage $f(i',j') = 1 + f(i,j)$. Else, run the listing algorithm described above to stage i. If wv^{-1} appears, exit with $wv^{-1} \simeq_R \lambda$. If not, construct a permutation representation of S_j (any concrete representation will do, provided that the group product is computable). Construct each mapping ϕ of X into S_j. Such a mapping ϕ can be tested for the homomorphism property by checking that for each $r_1 \cdots r_s \in R$, $\phi(r_1) \circ \cdots \circ \phi(r_s)$ is the S_j identity. Note also, that if ϕ is a homomorphism, $\phi(wv^{-1})$ can be computed in S_j. For each ϕ that is a homomorphism, test whether $\phi(wv^{-1})$ is not the S_j identity. If such a homomorphism is found, exit with $wv^{-1} \not\simeq_R \lambda$.

If exit is reached, the result is clearly correct. We argue that the process terminates. If $wv^{-1} \simeq_R \lambda$ this will eventually be discovered in the listing. Otherwise, since G is finite, at some stage $f(i,j)$, G will be isomorphic to a subgraph of S_j (Cayley's Theorem) and so some ϕ will be an isomorphism, hence $\phi(wv^{-1})$ cannot be the S_j identity. □

The size $|(X,R)|$ of a finite presentation (X,R) is defined as the number of bits required to store the list of symbols in X and words in R, *i.e.*, $(|X| + \sum_{w \in R} |w|) \log |X|$. In many cases, $|(X,R)|$ is much smaller than the order of the group it presents. For example, the presentation (X,R) with $X = \{x,y\}$ and $R = \{x^4y^{-3}, x^{-2}y^{-1}x^{-1}y^{-1}xyxyxy^{-1}xy\}$ presents a finite group of order $2^{10}3^9$ (see [7]) whereas, according to our definition, $|(X,R)| = 22$. This suggests that finite presentations may be useful as a way of achieving data compression for finite groups. The maximum data compression achievable for a finite group by using one of its finite presentations (X,R) is attained when $|(X,R)|$ is the minimum. If one has the Cayley table (or an equivalent explicit representation) of a finite group, then one can compute its minimum presentation size. The following algorithm demonstrates this. Enumerate by size all finite presentations. For each one, list as strings over the generators all elements of the group by solving the word problem. Since the group order is known, we can reject a presentation if there are too few or too many elements. If the presentation presents a group of the correct order, note that we have, in effect constructed its Cayley table. Now, it is straightforward by exhaustion to test whether two groups, given by their Cayley tables are isomorphic.

2 Vertex transitive graphs

An isomorphism from a graph $G = (V, E)$ to a graph $G' = (V', E')$ is a bijection $\sigma : V \to V'$ such that $(i, j) \in E$ iff $(\sigma(i), \sigma(j)) \in E'$. Two graphs are said to be isomorphic if there exists an isomorphism taking one graph to the other. Graph isomorphism can be characterized in terms of adjacency matrices. Two graphs G and G' are isomorphic iff there exists a permutation matrix P such that $P^{-1}M_G P = M_{G'}$, where M_G and M'_G are the adjacency matrices of G and G', respectively. An automorphism of a graph G is an isomorphism such that $P^{-1}M_G P = M_G$ for a certain permutation matrix P. The set $\mathrm{AUT}(G)$ of automorphisms of G forms a group under composition. A graph is said to be vertex transitive (VT) if for every pair i, j of vertices there exists $\sigma \in \mathrm{AUT}(G)$ such that $\sigma(i) = j$.

Let Γ be a group, and X a set of generators for Γ. We describe a labelled graph $C(\Gamma, X) = (\Gamma, E, \ell)$. The edge set E consists exactly of those (g, g') such that either $g' = g \cdot x$ or $g = g' \cdot x$, where $x \in X$ (in that case x is used as the label of the edge (g, g')). A graph is said to be a Cayley graph if it is isomorphic to $C(\Gamma, X)$ for some Γ and X. A Cayley graph is regular, with degree $|X|$. Also note that every Cayley graph is vertex transitive whereas the converse is not true (the Petersen graph is probably the best known example of a vertex transitive non-Cayley graph). A path $g_1, \ldots, g_r$ in $C(\Gamma, X)$ can be interpreted as the equation $g_1 \cdot x_1^{\sigma_1} \cdots x_{r-1}^{\sigma_{r-1}} = g_r$, where $\sigma_i = 1$ if $(g_i, g_{i+1} \cdot x_i) \in E$, otherwise $\sigma_i = -1$, *i.e.*, $(g_i, g_{i+1} \cdot x_i^{-1}) \in E$. This observation suggests a simple method of constructing the operation table for Γ from $C(\Gamma, X)$. Indeed, letting $w(g_1, g_r) = x_1^{\sigma_1} \cdots x_{r-1}^{\sigma_{r-1}}$, we have that $g_i \cdot g_j = \iota \cdot w(\iota, g_i) \cdot \iota \cdot w(\iota, g_j) = w(\iota, g_i) \cdot w(\iota, g_j)$.

We show how to construct $C(\Gamma, X)$ from a finite presentation (X, R) of Γ. The analysis is left to the reader.

1. Initialize the vertex set $V = \{\iota\} \cup X$. Recall that ι is the group identity. Initialize the labelled edge set $E = \{(\iota, x) \mid x \in X\}$.

2. For each vertex $v \in V$ and each $x \in X$, if neither $(v, v \cdot x)$ nor $(v, v \cdot x^{-1})$ belong to E, put $(v, v \cdot x)$ in E and $v \cdot x$ in V. Notice that we need the computability of the word problem to make these determinations.

3. Repeat Step 2 until no change occurs in E and V.

Note that by construction, one can read the edge label from the strings denoting the vertices.

As an example of a Cayley graph, we describe a family of Cayley graphs $G_n = C(S_n, X)$ where S_n is the symmetric group on $\{1, 2, ..., n\}$ and $X = \{(1, 2), (1, 2, ..., n)\}$. Each vertex of G_n has degree 2. The edge labels are the permutations $(1, 2)$ and $(1, 2, \dots n)$. The next theorem shows that the diameter of G_n is very small relative to the number of vertices, which is $n!$.

Theorem 4 *G_n has diameter $O(n^3)$.*

Proof : First, we state some facts that can be easily verified:

1. For the distinct numbers $k_1, k_2, \dots, k_m \in \{2, \dots, n\}$,
$$(1, k_1, \dots, k_m) = (1, k_1) \cdots (1, k_m) \ .$$
2. Furthermore,
$$(k_1, \dots, k_m) = (1, k_1) \cdots (1, k_m)(1, k_1) \ .$$
3. Next, for $1 < i \leq n$,
$$(1, i) = (1, 2)(2, 3) \cdots (i - 1, i)(i - 2, i - 1) \cdots (2, 3)(1, 2) \ .$$
4. Finally, for $0 < i < n$,
$$(i, i + 1) = (1, 2, \dots, n)^{-i+1}(1, 2)(1, 2, \dots, n)^{i-1} \ .$$

These facts show that (i) any permutation can be expressed as a product of $O(n^2)$ transpositions, and (ii) any transposition can be expressed as a product of $O(n)$ occurrences of $(1, 2)$ and $(1, 2, \dots, n)$. The theorem follows at once from (i) and (ii). □

3 The Sabidussi division problem

Let $G = (V, E)$ and m be a positive integer. The graph $mG = (V', E')$ is defined by $V' = V \times \{1, \dots, m\}$, and $((i, p), (j, q)) \in E'$ iff $(i, j) \in E$. It is easy to see that M_{mG} is the $m \times m$ block matrix, with each block being a copy of M_G. This justifies the description of mG in Step 1 of our scheme. The next result is due to G. Sabidussi [17].

Theorem 5 *If G is vertex transitive, there exists m dividing $|AUT(G)|$ such that mG is a Cayley graph.*

Let $\langle G\rangle$ denote the minimum compressed size of a graph G. Clearly, for a Cayley graph G, $\langle G\rangle$ is certainly no larger than the minimum over all $|(X,R)|$ such that $\Gamma(G)$ is isomorphic to $X^*/\simeq_R$. It follows from Theorem 5 that $\langle G\rangle$ of a vertex transitive graph is no larger than the minimum value of $\log(m)+\langle G'\rangle$, where G' is a Cayley graph isomorphic to mG. If G is a Cayley graph, we can take $m=1$, so $\log(m)=0$. There is the possibility that $m'G$ is also a Cayley graph for $m'>1$, and that $\langle m'G\rangle+\log(m')<\langle G\rangle$. We conjecture that this cannot happen. Recovering G up to isomorphism from a graph isomorphic to mG (by knowing m) will be called the Sabidussi division problem because one is 'dividing' by m.

It may happen that the Sabidussi division problem is ill posed in some cases. That is, there might exist non-isomorphic graphs G, G' such that mG and mG' are isomorphic for certain m. We conjecture that this does not happen. We come to our principal result.

Theorem 6 *If M_G is nonsingular, or excludes either 1 or -1 as an eigenvalue, then, up to isomorphism G can be recovered from m and a graph isomorphic to mG.*

Define $\tilde{G}$ to be the graph whose adjacency matrix is the $2n\times 2n$ matrix

$$\begin{pmatrix} M_G & I \\ I & M_G \end{pmatrix},$$

where I is the $n\times n$ identity matrix.

Conjecture 1 *Given $P^{-1}M_{\tilde{G}}P$, where P is a permutation matrix, G can be recovered up to isomorphism.*

This conjecture amounts to the assertion that if for some permutation matrix Q,

$$Q^{-1}M_{\tilde{G}}Q=\begin{pmatrix} A & I \\ I & A \end{pmatrix},$$

then G is isomorphic with the graph having adjacency matrix A. In the appendix we give a proof of this claim in the case when G is doubly vertex transitive (see Lemma 3 in the appendix). Assuming the Conjecture 1 holds we state and prove a couple of lemmas before turning to the proof of Theorem 6.

Lemma 1 *If G is vertex transitive, then the graph $\tilde{G}$ with adjacency matrix $M_{\tilde{G}}$ is vertex transitive.*

Proof : Label the vertices of $\tilde{G}$ in the obvious way as $1,\ldots,n,n+1,\ldots,2n$. If either $1\le i<j\le n$, or $n+1\le i<j\le 2n$, let σ be the automorphism of

G such that $\sigma(i) = j$ (or $\sigma(i-n) = j-n$). Then, let $\sigma' = \sigma$ on $\{1, \ldots, n\}$ and the identity on $n+1, \ldots, 2n$ (or, σ' is the identity on $1, \ldots, n$ and $\sigma'(k+n) = \sigma(k)$, for $k = 1, \ldots, n$). If $i \leq n$ and $n+1 \leq j$, let σ be the automorphism of G such that $\sigma(i) = j - n$. Let τ be the automorphism of $\tilde{G}$ (it is clearly an automorphism) such that τ swaps j and $j-n$. Then the required automorphism is $\tau \cdot \sigma'$, where $\sigma' = \sigma$ on $1, \ldots, n$ and the identity on $n+1, \ldots, 2n$. □

Lemma 2 *$M_{\tilde{G}}$ is nonsingular iff at least one of numbers* 1 *and* -1 *is not an eigenvalue of* M_G.

Proof : $M_{\tilde{G}}$ is singular iff there exists a vector x such that $M_{\tilde{G}}x = 0$. Write $x^T = (y, z)$, where y and z are n-dimensional. Singularity holds iff $M_G x = -y$ and $M_G y = -x$, *i.e.*, $(M_G)^2 x = x$, *i.e.*, 1 and -1 are eigenvalues of M_G. □

Proof : [Theorem 6] Assume M_G is nonsingular. First, we argue that if $P^{-1} O_m \otimes M_G P = O_m \otimes B$, and B is nonsingular, then $B = Q^{-1} M_G Q$, where Q is a permutation matrix.

Write P as an $m \times m$ block matrix with $n \times n$ blocks $R_{i,j}$ for $1 \leq i, j \leq m$. Note that P^{-1} is the $m \times m$ block matrix $R^T_{i,j}$. Now, $P^{-1} O_m \otimes M_G P$ is also an $m \times m$ block matrix with block $S_{i,j}$ being

$$(\sum_{k=1}^{m} R_{k,i})^T M_G \sum_{h=1}^{m} R_{h,j} \ .$$

We must have $B = S_{i,j}$, so both $(\sum_{k=1}^{m} R_{k,i})^T$ and $\sum_{h=1}^{m} R_{h,j}$ must be nonsingular. It is easy to see that this implies both must be permutation matrices. In particular, for $B = S_{1,1}$ we have that $(\sum_{k=1}^{m} R_{k,1})^T$ and $\sum_{h=1}^{m} R_{h,1}$ must be permutation matrices, but these two matrices are transposes of one another, and so inverses. That is, $B = X^{-1} M_G X$, where $X = \sum_{h=1}^{m} R_{h,j}$, which means B is the adjacency matrix of a graph isomorphic to G.

If M_G is singular, but excludes at least one of $1, -1$ as an eigenvalue, by Lemma 2, $M_{\tilde{G}}$ is nonsingular, and by Lemma 1, it is vertex transitive. We carry out the argument of the previous paragraph using $M_{\tilde{G}}$ in place of M_G. Thus, if $P^{-1} O_m \otimes M_{\tilde{G}} P = O_m \otimes B$, we can recover G up to isomorphism.

It remains to describe the overall algorithm. Let G' be either G or $\tilde{G}$, as appropriate. Given m and a finite presentation of $\Gamma(mG')$, (and the information as to whether G or $\tilde{G}$ was used in the compression), construct

mG' up to isomorphism (as explained in Section 2). That is, we actually have $P^{-1}M_{mG'}P$ for some permutation matrix P. Using m, loop through all permutation matrices Q of dimensions $mn \times mn$, if G was used, or $2mn \times 2mn$ if $\tilde{G}$ was used (note that n is now known), and if $Q^{-1}M_{mG'}Q = O_m \otimes B$, check whether B is nonsingular. If B is nonsingular, then it is the adjacency matrix of a graph isomorphic to either G of $\tilde{G}$ and we are done. Such a permutation matrix Q and matrix B must occur since we can take $Q = P$. □

4 Conclusion

In this paper we report an initial investigation of the compression of vertex transitive graphs using algebraic methods. Numerous problems remain to be solved several of which we mention here:

1. In the appendix we have shown that Conjecture 1 holds for doubly vertex transitive graphs. Does it also hold in the case of vertex transitive graphs?

2. How small is the minimum size presentation of an arbitrary vertex transitive graph? In other words, is it possible to quantify the compression efficiency of the proposed method?

3. Can each of the proposed steps be computed efficiently?

References

[1] M. Adler and M. Mitzenmacher. Towards compressing web graphs. In *Proceedings of Data Compression Conference*, 2001.

[2] G.E. Blelloch, B.M. Maggs, and S.L.M. Woo. Space-efficient finger search on degree-balanced search trees. In *Proceedings of the fourteenth annual ACM-SIAM symposium on Discrete algorithms*, 2003.

[3] P. Boldi and S. Vigna. The webgraph framework 1: Compression techniques. Technical Report 293-03, Universita di Milano, Dipartimento di Scienze dell'Informazione, 2003.

[4] S. Chen and J Reif. Efficient lossless compression of trees and graphs. In *Proceedings of Data Compression Conference*, 1996.

[5] Y. Chiang, C. Lin, and H. Lu. Orderly spanning trees with applications to graph encoding and graph drawing. In *Symposium on Discrete Algorithms*, 2001.

[6] N. Deo and B. Litow. A structural approach to graph compression. In *Proceedings of the MFCS'98 Workshop on Communication*, 1998.

[7] G. Havas, D. Holt, P. Kenne, and S. Rees. Some challenging group presentations. *Journal of the Australian Mathematical Society (Ser. A) 67*, pages 206–213, 1999.

[8] X. He, M. Kao, and H. Lu. A fast general methodology for information: Theoretically optimal encodings of graphs. In *European Symposium on Algorithms*, pages 540–549, 1999.

[9] M-C. Heydemann. Cayley graphs and interconnection networks. In G. Hahn and G. Sabidussi (eds.), Graph Symmetry: Algebraic Methods and Applications, NATO ASI Series C, vol. 497, 164-224. Kluwer, Dordrecht, 1997.

[10] K. Keeler and J. Westbrook. Short encodings of planar graphs and maps. *Discrete Applied Mathematics*, pages 239–252, 1995.

[11] H. Lu. Linear-time compression of bounded-genus graphs into information-theoretically optimal number of bits. In *Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms*, 2002.

[12] W. Magnus, A. Karras, and D. Solitar. *Combinatorial Group Theory, Presentation of Groups in Terms of Generators and Relations.* Dover Publications, 1976.

[13] Y.V. Matiyasevich. *Hilbert's Tenth Problem.* MIT Press, 1993.

[14] J.I. Munro and V. Raman. Succinct representation of balanced parentheses, static trees and planar graphs. In *IEEE Symposium on Foundations of Computer Science*, pages 118–126, 1997.

[15] S. Raghavan and H. Garcia-Molina. Representing web graphs. In *Proceedings of 19th IEEE International Conference on Data Engineering*, 2003.

[16] R.Albert and A-L.Barabasi. Statistical mechanics of complex networks. *Reviews of Modern Physics, (74)1*, pages 47–97, 2002.

[17] G. Sabidussi. Vertex-transitive graphs. *Monatsh. Math. 68*, pages 426–438, 1964.

[18] T. Suel and J. Yuan. Compressing the graph structure of the web. In *Proceedings of Data Compression Conference*, 2001.

Appendix

Proof : [Theorem 2] Let X be a generating set for the finite group Γ, and let R consist of all nonempty strings w over X of length at most $|\Gamma|$ such that $\prod w = \iota$. We claim (X, R) is a presentation of Γ. We prove this claim by showing that Γ is isomorphic to $X^*/\simeq_R$. Let w, v be two strings over X. The group product of the generators comprising w is denoted by $\prod w$. The group identity is denoted by ι, and the substring from i to j in w is denoted by $w[i, j]$.

First, we show that if $w \in X^*$ and $|w| > |\Gamma|$, then we can write $w = uzv$ where $z \in R$. By the pigeon hole principle, there exist $i < j$, with $j - i \leq |\Gamma|$ such that $\prod w[1, i] = \prod w[1, j]$. This implies that $\prod w[i+1, j] = \iota$ *i.e.*, $w[i+1, j] \in R$.

Next, we show that the mapping $\tau : X^*/\simeq_R \rightarrow \Gamma$ sending $[w]$ to $\prod w$ is well-defined. Assume that $w \simeq_R v$, and show that $\prod w = \prod v$. We induct on $|wv|$. The case $w = v = \lambda$ is trivial, since $\prod \lambda = \iota$. Assume $|w| > 0$. It again suffices to show that $wv^{-1} \simeq_R \lambda$ implies $\prod wv^{-1} = \iota$. If $0 < |wv| \leq |\Gamma|$, then $wv^{-1} \in R$, so $\prod wv^{-1} = \iota$. If $|wv| > |\Gamma|$, then we employ our observation to get $wv^{-1} = x'zx''$, where $z \in R$, so $wv^{-1} \simeq_R x'x'' \simeq_R \lambda$. Again, we proceed by induction since $|x'x''| < |wv|$.

Next, we prove that τ is injective. Assume $\prod w = \prod v$, and show that $w \simeq_R v$. It suffices to show that $wv^{-1} \simeq_R \lambda$. If $\prod w = \prod v$, then $\prod wv^{-1} = \iota$. If $|wv^{-1}| \leq |\Gamma|$, then $wv^{-1} \in R$, so $wv^{-1} \simeq_R \lambda$, *i.e.*, $w \simeq_R v$. Now, suppose that $|wv^{-1}| > |\Gamma|$. By the observation we made, $wv^{-1} = x'zx''$ such that $z \in R$. Now $\prod wv^{-1} = \prod x'x'' = \iota$. We can now proceed by induction since $|x'x''| < |wv^{-1}|$, *i.e.*, we have $x'x'' \simeq_R \lambda$.

Finally, it is straightforward to show that τ is a surjection and a homomorphism. □

Next, we prove a lemma which implies that the Conjecture 1 is true in the case when G is doubly vertex transitive.

Lemma 3 *Let G be a doubly vertex transitive graph with adjacency matrix M and let*

$$M_{\tilde{G}} = \begin{pmatrix} M & I \\ I & M \end{pmatrix}$$

If there is a permutation matrix Q such that

$$Q^{-1} M_{\tilde{G}} Q = \begin{pmatrix} A & I \\ I & A \end{pmatrix}$$

where A is the adjacency matrix of a doubly vertex transitive graph G_1 of degree at least 3, then G is isomorphic with G_1.

Proof : Recall that a graph is doubly vertex transitive if for any two pairs of vertices $(u, v), (w, x)$ there exists an automorphism f such that $f(u) = v$ and $f(w) = x$. Let

$$U = Q^{-1} \begin{pmatrix} 0 & I \\ I & 0 \end{pmatrix} Q$$

and

$$V = Q^{-1} \begin{pmatrix} M & 0 \\ 0 & M \end{pmatrix} Q$$

Note that U is a permutation matrix and V is the adjacency matrix of a two component graph, with each component being isomorphic to G. Also note that

$$U + V = \begin{pmatrix} A & I \\ I & A \end{pmatrix}$$

We prove now that

$$V = \begin{pmatrix} A & 0 \\ 0 & A \end{pmatrix}$$

Assume by contradiction that this is not true. Let $[n]$ be the vertex set of G and the matrices U, V are to be regarded as adjacency matrices of graphs with vertex set $[2n]$. The set of unordered pairs of elements of a set S is denoted by $S^{(2)}$. Let $N_1 \cup N_2$ be the partition of $[2n]$ such that one component of V has its edges in $N_1^{(2)}$ and the other has its edges in $N_2^{(2)}$. Note that A is the adjacency matrix of the subgraph of the graph corresponding to $U + V$ induced by the vertices in $[n]$. Let A_1, A_2 be the subgraphs of A induced by the $N_1 \cap [n]$ and $N_2 \cap [n]$, respectively. Then any edge (x, y) with $x \in A_1$ and $y \in A_2$ must be from U. Note that we can treat these edges of U as undirected, since if (x, y) is in U, then so must (y, x) because A is the adjacency matrix of an undirected graph. If (x, y) and (x, y') are any edges of U, then $y = y'$ because U is a permutation matrix. This implies that there exist $2r$ distinct vertices $x_1, x_2, ..., x_r \in A_1$ and $y_1, y_2, ..., y_r \in A_2$ such that $(x_i, y_i) \in U$ for all $i = 1, ..., r$ and these are the only edges between A_1 and A_2. Since A is 3-connected, it follows $r \geq 3$ (we only need $r \geq 2$ here). We argue that A can not be doubly vertex transitive. Since A is at least 3-regular, there exists $w \neq y_2$ in A_2 such that (y_1, w) is an edge of A_2. Let f be an automorphism such that $f(x_1) = x_2$ and $f(y_1) = w$. Then the edge (x_1, y_1) goes to (x_2, w) under f, but this is impossible since $w \neq y_2$. □