AMSI VACATION RESEARCH SCHOLARSHIPS 2018-2019

Free Products of Graphs

Max Carter

Supervised by George Willis & Stephan Tornier

University of Newcastle

Vacation Research Scholarships are funded jointly by the Department of Education and Training and the Australian Mathematical Sciences Institute.

AMSI AUSTRALIAN MATHEMATICAL SCIENCES INSTITUTE

Australian Government
Department of Education and Training

Abstract

In this project we are particularly interested in the idea of a free product of graphs. The free product of graphs is similar in a sense to the free product of groups. A free product of two groups G_1 and G_2 creates a new group $G_1 * G_2$ that consists of all possible 'reduced words' formed from the elements of the two groups. If X_1 and X_2 are the Cayley graphs of the two groups G_1 and G_2 respectively, then the Cayley graph of $G_1 * G_2$ is the free product of the two Cayley graphs X_1 and X_2. The new Cayley graph that is created from the free product of X_1 and X_2 is an infinite and highly symmetrical graph.

There are a number of different definitions of free products of graphs in the literature that define a free product between any arbitrary graphs; not just Cayley graphs associated with groups. In particular, there is a paper that discusses free products of rooted graphs and other definitions of free product of graphs that are not necessarily concerned with whether a graph is rooted or not. This paper compares these different definitions of free products of graphs alongside a new definition made in the theory of totally disconnected, locally compact groups. In the second half of the report we look at the automorphisms of free product graphs and provide a classification of the automorphisms of a free product graph contained in a particular subgroup of the automorphism group.

1 Introduction

Given two groups G_1 and G_2, we can form their free product $G_1 * G_2$ which is a group consisting of all possible sequences of elements from the original two groups. We call these sequences *words* and say that a word is *reduced* if we remove all occurrences of the identity element from the sequence and if two adjacent elements in the sequence are from the same group, we replace this pair with their product. The elements of the group $G_1 * G_2$ are equivalence classes of words, where two words are equivalent if one can be reduced to form the other word and the group operation is concatenation of words.

When working with two groups we can consider their Cayley graphs; a graph that contains all the elements of the group as vertices and an edge connects two vertices if we can multiply the group element associated with one vertex by a generator, from a predetermined generating set, to get the group element associated with the other vertex (See Appendix 1 for more detail). If G_1 and G_2 are two groups with Cayley graphs Γ_1 and Γ_2 respectively, the Cayley graph of the group $G_1 * G_2$ is a very interesting graph; it is an infinite, highly symmetric graph and contains a copy of both Γ_1 and

Γ_2 at each vertex. Now, what if we had a definition of 'free product' for graphs, where we could take some Cayley graphs $\Gamma_1, \ldots, \Gamma_n$ and form their free product which results in a graph isomorphic to the Cayley graph of the free product of their respective groups; and could we generalise such a definition to a free product of any arbitrary graphs? It just so happens there is, in fact there are a multitude of different definitions of free products of graphs in the literature, each one being vastly different from the other in their descriptions.

Some of these definitions of free products of graphs have large restrictions placed on their input graphs, while other definitions are more general and can be used to take the free product of almost any arbitrary graphs. In the sequel we will give an overview of the most prominent definitions of free products of graphs, followed by an analysis of the similarities between these definitions.

2 Free Products of Graphs

2.1 Overview of Current Definitions

Below we will list some of the main definitions of free products of graphs from the literature that we are concerned with in this paper. The first and main definition of free products of graphs that we will be working with is the following definition made by *George Willis* in the theory of totally disconnected, locally compact groups (see [1]).

Definition 1 [Willis]. Let $\Gamma_1, ..., \Gamma_n$ be connected graphs. Denote $V^{(l)}$ as the collection of all admissible strings of length l. Let $\mathbf{u} : V(\ast_{i=1}^{n}\Gamma_i) \to V(\Gamma_1) \times \cdots \times V(\Gamma_n)$ be a function which we will refer to as the update function. We define the update function and the collection of admissible strings inductively. First, begin by setting $V^{(0)} = \{\varnothing\}$ and $\mathbf{u}(\varnothing) = (u_1, ..., u_n) \in V_1 \times \cdots \times V_n$.

Then assume that $V^{(k)}$ and $\mathbf{u}\big|_{V^{(k)}}$ have been defined for all $k \in \{0, ..., l\}$. For any $\tilde{v} = v_1 \cdots v_k \in V^{(k)}$, denote $\mathbf{u}(\tilde{v}) = (u_1(\tilde{v}), ..., u_n(\tilde{v}))$, the string $\tilde{v}v_{k+1}$ is admissible if $v_k \in V_i$ and $v_{k+1} \in V_j$ with $i \neq j$ and $v_{k+1} \neq u_j(\tilde{v})$. The update function for this string is defined as:

$$u_h(\tilde{v}v_{k+1}) = \begin{cases} u_h(\tilde{v}) & \text{if } h \neq j \\ v_{k+1} & \text{if } h = j \end{cases}$$

Define the edges that are in the graph as follows. For the empty string, the edge $\{\varnothing, v_1\}$ is an edge if $\{u_i(\varnothing), v_1\}$ is an edge in the graph Γ_i. For an arbitrary admissible string $\tilde{v} = v_1, \ldots v_l$ in the free-product, an edge to this vertex can be formed in two different ways. First, construct an edge $\{\tilde{v}v_{l+1}, \tilde{v}v'_{l+1}\}$ (for $v_{l+1}, v'_{l+1} \in V(\Gamma_i)$) if $\{v_{l+1}, v'_{l+1}\}$ forms an edge in the graph Γ_i. Then the other type of edge we are allowed to form is an edge $\{\tilde{v}, \tilde{v}v_{l+1}\}$ if $v_{l+1} \in V(\Gamma_j)$, $\{u_j(\tilde{v}), v_{l+1}\} \in E(\Gamma_j)$. $\diamond$

The second definition we will be drawing comparisons with is from the unpublished manuscript *Amalgamated Free Products of Graphs and Arc-Types* by *Möller et al.* in [2] which defines a free product of graphs as follows:

Definition 2 [Möller–Seifter–Woess–Zemljič]. Let Γ_1 and Γ_2 be two connected vertex-transitive graphs with order (number of vertices in the graph) m and n respectively. Let $\mathcal{T}_{m,n}$ be the (m, n)-biregular tree with its vertex set naturally partitioned into the two sets V_1 and V_2 where every vertex in V_1 has degree m and every vertex in V_2 has degree n.

For each vertex $v \in V_1$ associate a copy Γ_v of the graph Γ_1 at each vertex and define a bijection ψ_v between the vertices of Γ_v and the edges incident to the vertex v in the graph $\mathcal{T}_{m,n}$. Similarly, for each vertex $w \in V_2$, we associate a copy Γ_w of the graph Γ_2 and bijection ψ_w. Then the free product $\Gamma_1 \ast \Gamma_2$ is formed by taking the union of all the graphs Γ_v such that $v \in V(\mathcal{T}_{m,n})$ and if $e = \{v, w\}$ is an edge in $\mathcal{T}_{m,n}$ such that $\psi_v(x) = e = \psi_w(y)$ for some $x \in X_v, y \in X_w$ then we identify the two vertices x and y. $\diamond$

The following two definitions are more concerned with the free product of rooted graphs, however, they are still of interest to our studies. The following definition is from a research paper *Growth in Products of Graphs* by Pisanki and Tucker (see [3]).

Definition 3 [Pisanski–Tucker]. Let G_r and H_s be rooted graphs with roots r and s respectively. The vertex set of the free product $G_r \ast H_s$ is the set of all finite sequences of vertices from $(V(G_r) - \{r\}) \cup (V(H_s) - \{s\})$ where the sequences alternate between $V(G_r) - \{r\}$ and $V(H_s) - \{s\}$. An edge joins two sequences α and β if and only if either $\alpha = \gamma$ and $\beta = \gamma x$ where x is a vertex adjacent to the corresponding root vertex or, $\alpha = \gamma x$ and $\beta = \gamma y$ with x, y being adjacent vertices in one of the graphs. $\diamond$

The fourth and final definition we will discuss in this section (and in this paper) is the following definition from the paper *Combinatorics of Free Product Graphs* by Gregory Quenell (see [4]). This definition is particularly aimed at the free product of Cayley graphs, however, it is obvious that this definition could be generalised to the free product of any connected vertex-transitive graphs as seen in definition 5 to come.

Definition 4 [Quenell]. Let Γ_1 and Γ_2 be two Cayley graphs of groups G_1 and G_2 respectively. Recursively define graphs B_1 and B_2 as follows:

- B_1 is the graph comprising of a single copy of Γ_1 with an independent copy of B_2 glued, by its identity vertex, to each of the non-identity vertices in the copy of Γ_1.
- B_2 is the graph comprising of a single copy of Γ_2 with an independent copy of B_1 glued, by its identity vertex, to each of the non-identity vertices in the copy of Γ_2.

The free product $\Gamma_1 * \Gamma_2$ is the graph constructed by taking the *connected sum* $B_1 \sharp B_2$ (see definition 2.6 in [4]) of the graphs B_1 and B_2. This is the graph formed by glueing B_1 and B_2 together by their identity vertices. ◇

Definition 4 is also extended to the free product of an arbitrary number of graphs as seen in definition 4.7 in [4] and outlined below:

Definition 5 [Quenell]. Let Γ_i be rooted graphs with roots e_i for $i = 1, \dots, k$. The branch B_i is the graph formed by glueing copies of every B_j, $j \neq i$, by the root vertices to each of the vertices in Γ_i except e_i. The free product graph $\Gamma_1 * \cdots * \Gamma_k$ is the graph $B_1 \sharp \cdots \sharp B_n$. ◇

Throughout this paper we will mainly be focusing on working with the first three definitions outlined here. The goal of this sections was to illustrate to the reader the wide variety of definitions of free products of graphs there are in the literature.

2.2 Do these definitions produce isomorphic graphs?

In this section we aim to prove that the definitions outlined in section 2.1 produce isomorphic graphs under some given restrictions. To avoid confusion between the definitions, for the remainder of section 2.2, we will let $*_1$ denote the free product of graphs using definition 1 from the previous section and similarly use $*_i$ to denote the free product using the i^{th} definition. We start with a short lemma which shows that the construction of a free product graph with definition 2 does not depend on the bijections used for the construction.

Lemma 2.2.1. Let Γ_1 and Γ_2 be connected vertex-transitive graphs with order m and n respectively. Let Λ_1 be the free product $\Gamma_1 \ast_2 \Gamma_2$ constructed using the set of bijections $\{\varphi_v : v \in V(\mathcal{T}_{m,n})\}$ and Λ_2 be the free product $\Gamma_1 \ast_2 \Gamma_2$ constructed using the bijections $\{\mu_v : v \in V(\mathcal{T}_{m,n})\}$. Then $\Lambda_1 \cong \Lambda_2$.

Proof (Sketch): For each copy of Γ_1 associated with a vertex $u \in V(\mathcal{T}_{m,n})$, let $V(\Gamma_u) = \{\xi_{u,1}, \dots, \xi_{u,m}\}$. Similarly for each copy of Γ_2 associated with a vertex $v \in V(\mathcal{T}_{m,n})$, let $V(\Gamma_v) = \{\xi_{v,1}, \dots, \xi_{v,n}\}$. Also label each of the vertices in Λ_1 by the edge it is associated with in the underlying tree and label each of the vertices in Λ_2 the same except with a 'prime' this time. Now, define $\psi : \Lambda_1 \longrightarrow \Lambda_2$ inductively as follows. Choose some vertex $\epsilon_1 \in V(\Lambda_1)$; then there exists some adjacent $u, v \in V(\mathcal{T}_{m,n})$ such that the vertex ϵ_1 is a result of identifying two vertices, one from Γ_u and one from Γ_v. Let $\epsilon_2 \in V(\Lambda_2)$ be the vertex identified between the two graphs Γ_u and Γ_v in Λ_2; define $\psi(\epsilon_1) = \epsilon_2$.

Then suppose that ψ has been defined on all vertices in Λ_1 of up to distance k from the vertex ϵ. Then given a vertex $x \in V(\Lambda_1)$ at distance $k+1$ from ϵ, define $\psi(x)$ as follows. Let $x' \in V(\Lambda_1)$ be a vertex at distance k from ϵ adjacent to x and suppose that $\psi(x') = y' \in V(\Lambda_2)$. The vertices x and x' are contained in some unique sheet, say Γ_u which is a copy of the graph Γ_i ($i = \{1, 2\}$). Then suppose that Γ_v is the copy of Γ_i attached at $y' \in V(\Lambda_2)$. Suppose that $\varphi_u^{-1}(x') = \xi_{u,k}$ and $\mu_v^{-1}(y') = \xi_{v,l}$. Then since Γ_i is vertex-transitive, there exists an automorphism $\alpha \in \mathrm{Aut}(\Gamma_i)$ such that $\alpha(\xi_{u,k}) = \xi_{v,l}$. The define $\psi(x) = \mu_v(\alpha(\xi_{u,j}))$ where $\varphi_u^{-1}(x) = \xi_{u,j}$.

The fact that ψ preserves adjacencies follows from α being an isomorphism. Injectivity and surjectivity of ψ follows by another two induction argument similar to that seen in the proof of proposition 2.2.2 that follows. $\square$

The proof to the following proposition displays an isomorphism between definitions 1 and 2 of free products of graphs. We assume that the graphs are vertex-transitive as there are cases when the two free product graphs will not be isomorphic if the graphs are not vertex-transitive. This issue will be discussed in more detail in section 2.3.

Proposition 2.2.2. If Γ_1 and Γ_2 are connected vertex-transitive graphs, then $\Gamma_1 \ast_1 \Gamma_2 \cong \Gamma_1 \ast_2 \Gamma_2$.

Proof: Let Γ_1 and Γ_2 be connected vertex-transitive graphs with order m and n respectively. Let $V(\Gamma_1) = \{\omega_1, \dots, \omega_m\}$ and $V(\Gamma_2) = \{\nu_1, \dots, \nu_n\}$. We need to construct an isomorphism $\psi : \Gamma_1 \ast_1 \Gamma_2 \longrightarrow \Gamma_1 \ast_2 \Gamma_2$; this will be done inductively. First define a graph embedding $\varphi_{(x,i)} : \Gamma_i \rightarrow \Gamma_1 \ast_2 \Gamma_2$ such that $\varphi_{(x,i)}$ is an isomorphism between the graph Γ_i and the unique Γ_i-sheet at the vertex $x \in V(\Gamma_1 \ast_2 \Gamma_2)$. Initiate the update function for constructing $\Gamma_1 \ast_1 \Gamma_2$ by $\mathbf{u}(\varnothing) = (\omega_a, \nu_b)$,

$a \in \{1, \ldots, m\}$ and $b \in \{1, \ldots, n\}$. Then choose some vertex $x_0 \in V(\Gamma_1 \circledast_2 \Gamma_2)$ such that $\varphi^{-1}_{(x_0,1)}(x_0) = \omega_a$ and $\varphi^{-1}_{(x_0,2)}(x_0) = \nu_b$; define $\psi(\varnothing) = x_0$. If there does not exist such an x_0, we can modify our choices for our initial values of $\mathbf{u}(\varnothing)$ to suit as this does not affect the structure of our graph (see Proposition 2.8 in [1]).

Now suppose that ψ has been defined on all admissible strings in $\Gamma_1 \circledast_1 \Gamma_2$ of up to length l. Given some $\tilde{v} = v_1 \cdots v_l \in V^{(l)}$ with $\psi(\tilde{v}) = y \in V(\Gamma_1 \circledast_2 \Gamma_2)$, if $\tilde{v}v_{l+1} \in V^{(l+1)}$, define $\psi(\tilde{v}v_{l+1})$ as follows. Suppose $v_l \in V(\Gamma_1)$ and that $u_2(\tilde{v}) = \nu_i$. If $\varphi^{-1}_{(y,2)}(y) = \nu_k$ and $v_{l+1} = \nu_j \in V(\Gamma_2)$ then we have $\{\nu_i, \nu_j\} \in E(\Gamma_2)$ by definition of the free product $\circledast_1$. Since Γ_1 and Γ_2 are both vertex-transitive graphs, there exists an $\alpha \in \mathrm{Aut}(\Gamma_2)$ such that $\alpha(\nu_i) = \nu_k$. Then define $\psi(\tilde{v}v_{l+1}) = \varphi_{(y,2)}(\alpha(\nu_j))$. Similarly, if $v_l \in V(\Gamma_2)$, define ψ in exactly the same way except with the roles of Γ_1 and Γ_2 exchanged as well as the ω's and ν's.

Now, if $\tilde{v} = v_1 \cdots v_l \in V^{(l)}$ and $\tilde{v}v_{l+1} \in V^{(l+1)}$ are two adjacent admissible string in $\Gamma_1 \circledast_1 \Gamma_2$ as defined in the previous paragraph, then $\alpha(\nu_i) = \nu_k$ and $\alpha(\nu_j)$ are adjacent in Γ_2 since α is a graph automorphism and it follows that $\psi(\tilde{v})$ and $\psi(\tilde{v}v_{l+1})$ are adjacent since φ is a graph isomorphism. Again, a similar argument can be applied for the case when $v_l \in V(\Gamma_2)$. Thus ψ is a graph homomorphism.

We need to show that this homomorphism is actually an isomorphism. To prove that it is injective, suppose $\tilde{v} = v_1 \cdots v_m \in V^{(m)}$ and $\tilde{v}' = v'_1 \cdots v'_n \in V^{(n)}$ with $\psi(\tilde{v}) = \psi(\tilde{v}')$. Let $\psi(\varnothing) = x_0 \in V(\Gamma_1 \circledast_2 \Gamma_2)$. We must have that $v_1, v'_1 \in V(\Gamma_i)$ and $\varphi_{(x_0,i)}(v_1) = \varphi_{(x_0,i)}(v'_1)$ otherwise $\psi(\tilde{v})$ and $\psi(\tilde{v}')$ would be in different branches of the free product graph $\Gamma_1 \circledast_2 \Gamma_2$. Since $\varphi_{(x_0,i)}$ is an isomorphism, $v_1 = v'_1$. Continuing by induction, with a similar argument we see that $m = n$ and $v_k = v'_k$ for all $k \in \{1, \ldots, m = n\}$ so that $\tilde{v} = \tilde{v}'$.

Similarly, we will prove surjectivity by induction. Suppose that $\psi(\varnothing) = x_0 \in V(\Gamma_1 \circledast_2 \Gamma_2)$. Clearly all the vertices in the Γ_1-sheet and Γ_2-sheet attached to x_0 have vertices mapping onto them. Then suppose that all Γ_i-sheets, $i = 1$ or 2, of up to distance k from x_0 have an element of $V(\Gamma_1 \circledast_1 \Gamma_2)$ mapping onto them. Given some Γ_i-sheet $\mathcal{S}$ at distance $k+1$ from x_0, there is a unique Γ_j-sheet $\mathcal{S}'$ $(i \neq j)$ at distance k from x_0 that $\mathcal{S}$ is attached to. Let x be the unique vertex shared by $\mathcal{S}$ and $\mathcal{S}'$; by hypothesis there exists $\tilde{v} \in V^{(l)}$ such that $\psi(\tilde{v}) = x$. It follows that given any $y \in V(\mathcal{S})$, there exists $v_{l+1} \in \Gamma_i$ such that $\psi(\tilde{v}v_{l+1}) = y$ since $\varphi_{(x,i)}$ is a graph embedding of Γ_i at the vertex $x \in V(\Gamma_1 \circledast_2 \Gamma_2)$. □

The following proposition displays an isomorphism between definition 1 and 3. We define a *rooted* graph to be a graph that has a distinguished vertex i.e. we choose one vertex from the graph and call it the 'root'. A rooted graph is vertex-transitive if when we disregard the root vertex, the underlying graph is vertex-transitive.

Proposition 2.2.3. If Γ_1 and Γ_2 are rooted connected vertex-transitive graphs, then $\Gamma_1 \ast_1 \Gamma_2 \cong \Gamma_1 \ast_3 \Gamma_2$.

Proof: Let Γ_1 and Γ_2 be rooted connected vertex-transitive graphs with roots r and s respectively. We will construct an isomorphism $\psi : \Gamma_1 \ast_1 \Gamma_2 \to \Gamma_1 \ast_3 \Gamma_2$ inductively. Initiate the update function for constructing the graph $\Gamma_1 \ast_1 \Gamma_2$ by $\mathbf{u}(\varnothing) = (r, s)$. Define $\psi(\varnothing) = \epsilon$ where ϵ is the empty string in the graph $\Gamma_1 \ast_3 \Gamma_2$.

Suppose that ψ has been defined on all admissible strings in $\Gamma_1 \ast_1 \Gamma_2$ of up to length l. Take some $\tilde{v} \in V^{(l)}$ with $\mathbf{u}(\tilde{v}) = (u_1(\tilde{v}), u_2(\tilde{v}))$ and suppose that $\psi(\tilde{v}) = x \in V(\Gamma_1 \ast_3 \Gamma_2)$. Then given $\tilde{v}v_{l+1} \in V^{(l+1)}$ with $v_{l+1} \in V(\Gamma_i)$, define $\psi(\tilde{v}v_{l+1}) = x\alpha(v_{l+1})$ where $\alpha \in \mathrm{Aut}(\Gamma_i)$ such that $\alpha(u_i(\tilde{v}))$ is the root vertex in the graph Γ_i.

This defines our map ψ. We need to prove that it is in fact an isomorphism. First we will show that ψ preserves adjacencies. There are two cases:

Case 1 - $\tilde{v}$ and $\tilde{v}v_{l+1}$ are adjacent in $\Gamma_1 \ast_1 \Gamma_2$: Suppose that $\tilde{v} \in V^{(l)}$ and $\tilde{v}v_{l+1} \in V^{(l+1)}$ are two adjacent vertices in $\Gamma_1 \ast_1 \Gamma_2$ and $\psi(\tilde{v}) = x \in V(\Gamma_1 \ast_3 \Gamma_2)$. Then $u_i(\tilde{v})$ is adjacent to v_{l+1} in the graph Γ_i. If $\alpha \in \mathrm{Aut}(\Gamma_i)$ is the automorphism such that $\alpha(u_i(\tilde{v}))$ is the root vertex, then $\alpha(v_{l+1})$ is adjacent to the root vertex since α is an automorphism and hence x and $x\alpha(u_i(\tilde{v}))$ are connected.

Case 2 - $\tilde{v}v_{l+1}$ and $\tilde{v}v'_{l+1}$ are adjacent in $\Gamma_1 \ast_1 \Gamma_2$:

Suppose that $\tilde{v}v_{l+1}, \tilde{v}v'_{l+1} \in V^{(l+1)}$ are two adjacent vertices in $\Gamma_1 \ast_1 \Gamma_2$ with $v_{l+1}, v'_{l+1} \in V(\Gamma_i)$ and $\psi(\tilde{v}) = x \in V(\Gamma_1 \ast_3 \Gamma_2)$. Then if $\alpha \in \mathrm{Aut}(\Gamma_i)$ is the automorphism such that $\alpha(u_i(\tilde{v}))$ is the root vertex in Γ_i, then $\alpha(v_{l+1})$ and $\alpha(v'_{l+1})$ are both adjacent vertices in Γ_i since α is an automorphism and hence $x\alpha(v_{l+1})$ and $x\alpha(v'_{l+1})$ are adjacent in $\Gamma_1 \ast_3 \Gamma_2$.

Now, to prove that ψ is injective, suppose that $\tilde{v} = v_1 \cdots v_m$ and $\tilde{v}' = v'_1 \cdots v'_n$ are two admissible strings in $\Gamma_1 \ast_1 \Gamma_2$ such that $\psi(\tilde{v}) = \psi(\tilde{v}')$. Clearly $m = n$ otherwise we could not have $\psi(\tilde{v}) = \psi(\tilde{v}')$ under this construction. Now we must have that $v_1 = v'_1$ otherwise $\psi(v_1) \neq \psi(v'_1)$ which would result in $\psi(\tilde{v})$ and $\psi(\tilde{v}')$ being located on different branches of the free product graph $\Gamma_1 \ast_3 \Gamma_2$. Continuing by induction, with a similar argument, we see that $v_k = v'_k$ for all $k \in \{1, \ldots, m = n\}$ and hence ψ is one-to-one.

We prove surjectivity also via induction. Clearly $\psi(\varnothing) = \epsilon$ and all vertices whose associated sequences are of length 1 in $\Gamma_1 *_3 \Gamma_2$ have a vertex in $\Gamma_1 *_1 \Gamma_2$ mapping to them. Now suppose that all vertices in $\Gamma_1 *_3 \Gamma_2$ associated with sequences of up to length l have a vertex from $\Gamma_1 *_1 \Gamma_2$ mapping onto them. Take some sequence $x \in V(\Gamma_1 *_3 \Gamma_2)$ of length $l+1$ and let x' be the sequence in $V(\Gamma_1 *_3 \Gamma_2)$ of length l whose first l entries are the same as x. Then, under assumption, there exists $\tilde{v} = v_1 \cdots v_l \in V^{(l)}$ such that $\psi(\tilde{v}) = x'$. Suppose $v_l \in V(\Gamma_j)$. There exists $\alpha \in \mathrm{Aut}(\Gamma_i)$ such that $\alpha(u_i(\tilde{v}))$ $(i \neq j)$ is the root vertex in Γ_i and $v_{l+1} \in V(\Gamma_i)$ such that $\alpha(v_{l+1})$ is the last member of the sequence x. It follows that $\psi(\tilde{v}v_{l+1}) = x$. □

The following proposition is the final proposition of this section. It states that definition 1 and 4 produce isomorphic graphs if the initial graphs are vertex-transitive. We omit the proof however as it is an almost identical argument to that of the proof in Proposition 2.2.2.

Proposition 2.2.4. If Γ_1 and Γ_2 are connected vertex-transitive graphs, then $\Gamma_1 *_1 \Gamma_2 \cong \Gamma_1 *_4 \Gamma_2$.

Proof (Sketch): Define $\psi : \Gamma_1 *_1 \Gamma_2 \longrightarrow \Gamma_1 *_4 \Gamma_2$ inductively. Let e be the vertex identified when taking the connected sum of B_1 and B_2 in the construction of $\Gamma_1 *_4 \Gamma_2$; define $\psi(\varnothing) = e$. Then there is a unique Γ_1-sheet and Γ_2-sheet attached to $\varnothing \in V(\Gamma_1 *_1 \Gamma_2)$, map these sheets onto the corresponding sheets attached to $e \in V(\Gamma_1 *_4 \Gamma_2)$.

Now suppose that ψ has been defined on all sheets of up to distance k from $\varnothing \in V(\Gamma_1 *_1 \Gamma_2)$. Then given a sheet $\mathcal{S} \in \Gamma_1 *_1 \Gamma_2$ at distance k from $\varnothing$ and the corresponding sheet $\mathcal{S}' \in \Gamma_1 *_2 \Gamma_2$ that it is mapped to by ψ, map each of the $|V(\mathcal{S})| - 1$ sheets attached to $\mathcal{S}$ at distance $k+1$ from $\varnothing$ onto the $|V(\mathcal{S})| - 1$ sheets attached to $\mathcal{S}'$ at distance $k+1$ from e in $\Gamma_1 *_4 \Gamma_2$. □

Corollary 2.2.5. If $\Gamma_1, \ldots, \Gamma_n$ are connected vertex-transitive graphs, then $\Gamma_1 *_1 \cdots *_1 \Gamma_n \cong \Gamma_1 *_5 \cdots *_5 \Gamma_n$.

Proof: Follows from the associativity of the free product. □

2.3 Discussion of the Definitions

2.3.1 Definition 2

The first definition in this paper of free products of graphs is the most general of the four definitions listed; it can take in any countable number of graphs and only requires them to be connected. This is not the case however in definition 2 (and definitions 3 and 4 for the matter). Definition 2 can only take an input of two graphs and it requires these graphs to be connected and vertex-transitive.

When working with definition 1 to construct a free product of two graphs, the construction of the free product graph begins at the empty string by initiating the update function, then iteratively defines all the admissible strings that act as vertices in the graph and the update functions that go along with each vertex/admissible string. The edges are then formed under a certain criteria as outlined earlier in section 2.1. After going through a few examples of constructing graphs with definition 1, it becomes apparent that there is a more geometric intuition as to what this construction is doing. If we have connected graphs $\Gamma_1, \ldots, \Gamma_n$ and construct the free product graph $*_{k=1}^{n}\Gamma_k$ using definition 1, we start by attaching all the graphs Γ_j $(j \in \{1, \ldots, n\})$ to the empty string by the vertex $u_j(\varnothing) \in V(\Gamma_j)$. Then, given an admissible string $\tilde{v} = v_1 \cdots v_l \in V^{(l)}$ in $*_{k=1}^{n}\Gamma_k$, the unique Γ_j-sheet at $\tilde{v}$ is attached to $\tilde{v}$ by the vertex $u_j(\tilde{v}) \in V(\Gamma_j)$. Essentially, given a particular Γ_j-sheet in the free product graph, we attach all the graphs Γ_i with $i \neq j$ to each of the vertices in the Γ_j-sheet that don't already have a copy of Γ_i attached. The update function provides the information as to which vertex in Γ_i is attached to the Γ_j-sheet.

This particular construction of the free product graph has already somewhat 'predetermined' what orientation each of the graphs $\Gamma_1, \ldots, \Gamma_n$ are going to attach to the admissible strings in $*_{k=1}^{n}\Gamma_k$. This is not the case however in definition 2; there is some ambiguity in what orientation each of the graphs are going to be connected to each other. In definition 2, if both of the graphs that you are constructing the free product of are not vertex-transitive graphs, then the resultant free product graph may not be isomorphic to the free product of the graphs if definition 1 is used, as the orientation of the graphs at each vertex is random and does not have 'predefined' what orientation each of the graphs will connect to each other.

In saying that however, even if both of the graphs are not vertex-transitive, a connected graph will still be created, it just won't necessarily be isomorphic to the graph constructed if definition 1 is used. This issue could be remedied however; we could more precisely define how each of the vertices will be identified during the construction of the free product graph by labelling all the vertices of the two graphs and explicitly define what vertices can be identified with each other and which cannot in a certain way so that the 'correct' graph will be constructed. This will result is a construction akin to that of definition 1.

What has been said in the previous paragraphs would seem to indicate that definition 1 is a far more superior construction for the free product of graphs than definition 2. Whilst this may be the case in certain instances, definition 2 provides a very intuitive and easy construction of the free product of two graphs, allowing for a quick visualisation of what the resultant graph will look like. On the other hand, definition 1 is more complicated and takes more time to construct a graph, however it provides a more precise construction which is favourable in certain circumstances.

2.3.2 Definition 3

The construction of definition 3 by Pisanski and Tucker is very similar to the construction in definition 1 by Willis, however, there is one distinct difference.

Both definitions 1 and 3 start by defining all the vertices in the free product graph to be sequences of vertices from the base graphs and they provide some criteria as to which sequences will be 'allowed' to form vertices in the free product graph. The two definitions also define the edge relations in an almost identical way.

There is one major difference between these definitions however. As described in section 2.3.1, definition 1 intuitively constructs the free product graph by connecting all the graphs to the empty string by the vertex in the update function on the empty string, and then continues to construct the free product graph by joining each of the graphs to each of the admissible strings that do not already contain a copy of that graph and the update function determines the orientation at which the graph attaches. In comparison, in definition 3 there is no update function, the free product graph is constructed by attaching each graph by its root vertex each time it is placed into the free product graph during the construction. This means that in definition 3 the graphs attach at each of the strings with the same orientation each time whereas in definition 1 the 'orientation' varies depending on the admissible string it is attaching to.

This creates the problem we encountered when trying to prove that the two definitions produce isomorphic graphs. If the two graphs that we are taking the free product of are not vertex-transitive, then the differences in how the graphs attach at each of the strings in the free product graph may result in the two definitions producing non-isomorphic graphs.

2.3.3 Definition 4

Definition 4 is similar in a sense to definition 2. While definitions 1 and 3 both construct the free product graph by labelling vertices with sequences of vertices from the original graphs and then defining edge relations in a similar method, definitions 2 and 4 build the free product graph with a more 'visual' construction rather than specifically defining vertices and edge relations.

Given two connected vertex-transitive graphs Γ_1 and Γ_2, both definitions construct the free product graph essentially by attaching a copy of Γ_1 to each of the vertices in Γ_2 and similarly attaching a copy of Γ_2 to each of the vertices in the copies of Γ_1 just attached and repeating this process indefinitely. Both of the definitions require that the graphs are vertex transitive as they do not specify the orientation as to which each of the sheets in the free product graph need to attach unlike definitions 1 and 3.

3 Automorphisms of Free Product Graphs

3.1 Classification of Automorphisms of a Free Product Graph

In Jacque Tits' paper *Sur le groupe des automorphismes d'un arbre* ([5]), he classifies automorphisms of a tree into three classes; elliptic, inversion and hyperbolic automorphisms. Elliptic automorphisms are automorphisms that fix a vertex, inversions are automorphisms that invert an edge and hyperbolic automorphisms are essentially any automorphism that does not fall into either of these classes. To classify the automorphisms, Tits' defines a function on the automorphisms of the tree which outputs the minimum distance any point in the tree is moved under a given automorphism. This construction allows him to classify the automorphisms in this particular way. In the sequel we will begin to develop a classification for automorphisms of graphs constructed from a free product of graphs similar to that of trees in Tits' paper.

Before we get started, there are a few terms that need to be defined first. As in [1], given a path $\mathcal{P} = \{x_1, \ldots, x_n\} \subset V(*_{i=1}^n \Gamma_i)$, a point x_k in the path is called a *transition point* of the path $\mathcal{P}$ if $\{x_{k-1}, x_k\} \in E(\Gamma_i)$ and $\{x_k, x_{k+1}\} \in E(\Gamma_j)$ for $i \neq j$. A path $\mathcal{P}_{[\tilde{v},\tilde{v}']}$ between two vertices $\tilde{v}, \tilde{v}' \in V(*_{i=1}^n \Gamma_i)$ will be called *minimal* if there is no other distinct path between $\tilde{v}$ and $\tilde{v}'$ of strictly shorter length. The path $\mathcal{P}_{[\tilde{v},\tilde{v}']}$ is called *reduced* if there are no cycles or backtracking in the path.

Let $(\mathcal{S}_j)_{j\in J}$ be a sequence of sheets in the free product graph $\ast_{i=1}^n \Gamma_i$ with either $J = \mathbb{Z}$, $J = \mathbb{N}$ or $J = [0, n] \subset \mathbb{N} \cup \{0\}$. Then we say that $(\mathcal{S}_j)_{j\in J}$ is a *string* of sheets in $\ast_{i=1}^n \Gamma_i$ if it satisfies the following conditions:

- $\mathcal{S}_j$ is connected to $\mathcal{S}_{j+1}$ in $\ast_{i=1}^n \Gamma_i$ for each $j \in J$.
- For each $k \in J$, there are at most two sheets in $\{\mathcal{S}_j\}_{j\in J}$ attached to $\mathcal{S}_k$
- $\mathcal{S}_{j-1}$ and $\mathcal{S}_{j+1}$ do not share the same vertex in $\mathcal{S}_j$ for each $j \in J$.

If $J = \mathbb{N}$, then we call $(\mathcal{S}_j)_{j\in J}$ an infinite string of sheets in $\ast_{i=1}^n \Gamma_i$. If $J = \mathbb{Z}$, then we call $(\mathcal{S}_j)_{j\in J}$ a bi-infinite string of sheets in $\ast_{i=1}^n \Gamma_i$. Let $[\mathcal{S}, \mathcal{S}']$ denote the unique string between the sheets $\mathcal{S}, \mathcal{S}' \in \ast_{i=1}^n \Gamma_i$. In this case we will call $\mathcal{S}$ and $\mathcal{S}'$ the initial and terminal sheets respectively. Given a string of sheets $[\mathcal{S}, \mathcal{S}']$, we define the length of this string to be the number of sheets in the string and denote the length by $\ell(\mathcal{S}, \mathcal{S}')$.

3.1.1 Classification of Automorphisms via Their Action on Sheets

Let $\Gamma_1, \ldots, \Gamma_n$ be connected graphs and let $\mathcal{S}$ denote a sheet in the free product graph $\ast_{i=1}^n \Gamma_i$. Given two sheets $\mathcal{S}$ and $\mathcal{S}'$, define the *distance* between the two sheets, $\mathfrak{d}(\mathcal{S}, \mathcal{S}')$, to be $\mathfrak{d}(\mathcal{S}, \mathcal{S}') = \ell(\mathcal{S}, \mathcal{S}') - 1$. Now, lets restrict our attention to the subgroup $G \leqslant \mathrm{Aut}(\ast_{i=1}^n \Gamma_i)$ that contains only automorphism that send sheets to other sheets in the free product graph.

Given any $\alpha \in \mathrm{Aut}(\ast_{i=1}^n \Gamma_i)$, define a function $\|\cdot\| : \mathrm{Aut}(\ast_{i=1}^n \Gamma_i) \longrightarrow \mathbb{N} \cup \{0\}$ by:

$$\|\alpha\| = \min\{\mathfrak{d}(\mathcal{S}, \alpha(\mathcal{S})) : \mathcal{S} \in \ast_{i=1}^n \Gamma_i\}$$

Using the above function, it is now possible to classify the automorphisms in this particular subgroup of the automorphism group into classes depending on how they act on individual sheets in the free product graph.

If $\alpha \in G$ such that $\|\alpha\| = 0$, then the automorphism α is stabilising some sheet $\mathcal{S}$ in the free product graph i.e. $\alpha(\mathcal{S}) = \mathcal{S}$. Notice that it does not necessarily have to fix all the vertices of $\mathcal{S}$ though. The automorphisms that satisfy this condition include all the automorphisms of the component graphs acting as a symmetry about a single sheet in the free product graph, provided such an automorphism exists of course.

Those automorphisms $\alpha \in G$ satisfying $\|\alpha\| = 1$ fall into two classes. Either α permutes adjacent isomorphic sheets in the free product graph, while stabilising no single sheet in the graph or α is a non-trivial translation along a bi-infinite string of sheets in the free product graph. As we will see in the theorem to follow, these particular automorphism can be represented by the composition of an automorphism that fixes a vertex with an automorphism that stabilises a sheet in the free product graph.

Automorphisms satisfying $\|\alpha\| > 1$ are analogous to hyperbolic automorphisms of a regular tree. These automorphisms stabilise no sheet in the graph; we can think of these automorphisms as being a non-trivial translation along a bi-infinite string of sheets in the free product graph.

The Theorem 3.1.3 below provides a classification of automorphisms that are contained in this subgroup of automorphisms that send sheets to other sheets in the free product graph. Before we get to the theorem, we have two lemma's.

Lemma 3.1.1. Let $\Gamma_1, \ldots, \Gamma_n$ be connected graphs and suppose that there exists an isomorphism $\varphi : \Gamma_1 \longrightarrow \Gamma_2$. If $\mathcal{S}$ and $\mathcal{S}'$ are the sheets in $*_{i=1}^n \Gamma_i$ associated with Γ_1 and Γ_2 respectively attached to $\varnothing$ with $\varphi(u_1(\varnothing)) = u_2(\varnothing)$, then there exists an automorphism $\alpha \in \mathrm{Aut}(*_{i=1}^n \Gamma_i)$ such that $\alpha(\mathcal{S}) = \mathcal{S}'$, $\alpha(\mathcal{S}') = \mathcal{S}$ and α fixes $\varnothing$.

Proof: Assume the hypotheses. Let $\alpha : \Gamma_1 * \Gamma_2 \longrightarrow \Gamma_1 * \Gamma_2$. Begin by defining $\alpha(\varnothing) = \varnothing$. Then let $\psi(v) = \varphi(v)$ if $v \in V(\Gamma_1)$, $\psi(v) = \varphi^{-1}(v)$ if $v \in V(\Gamma_2)$ and $\psi(v) = v$ if $v \notin V(\Gamma_1) \cup V(\Gamma_2)$. Define

$$\alpha(\tilde{v}) = \begin{cases} \psi(v_1) \cdots \psi(v_l), & \text{if } v_1 \in V(\Gamma_1) \cup V(\Gamma_2) \\ v_1 \cdots v_l, & \text{if } v_1 \notin V(\Gamma_1) \cup V(\Gamma_2) \end{cases}$$

for any $\tilde{v} = v_1 \cdots v_l \in V^{(l)}$.

First we need to show that for any $\tilde{v} = v_1 \cdots v_l \in V^{(l)}$, $\psi(v_1) \cdots \psi(v_l) \in V^{(l)}$. Clearly if v_i, v_j are elements of the string not from the same graph, then $\psi(v_i), \psi(v_j)$ are also vertices not from the same graph by definition of ψ and hence the first condition for admissible strings is satisfied. Then if $\tilde{v}, \tilde{v}v_{l+1}$ are admissible strings, assuming that $v_{l+1} \in V(\Gamma_i)$, $\psi(u_i(\tilde{v})) \neq \psi(v_{l+1})$ since φ and φ^{-1} are isomorphisms thus the second condition for admissible strings is satisfied.

Now we need to prove that α is an automorphism. To prove that α preserves adjacencies, there are two cases:

Case 1 ($\tilde{v}, \tilde{v}v_{l+1}$ are adjacent): Suppose that $\tilde{v} = v_1 \cdots v_l$ and $\tilde{v}v_{l+1}$ are adjacent admissible strings. Then supposing that $v_{l+1} \in V(\Gamma_i)$, $\psi(v_{l+1})$ is adjacent to $\psi(u_i(\tilde{v}))$ since φ and φ^{-1} are isomorphisms and $u_i(\tilde{v}), v_{l+1}$ are adjacent, thus $\psi(v_1)\psi(v_2)\cdots\psi(v_l)$ and $\psi(v_1)\psi(v_2)\cdots\psi(v_l)\psi(v_{l+1})$ are adjacent.

Case 2 ($\tilde{v}v_{l+1}, \tilde{v}v'_{l+1}$ are adjacent): Suppose that $\tilde{v}v_{l+1}, \tilde{v}v'_{l+1}$ are adjacent admissible strings. Then if $v_{l+1}, v'_{l+1} \in V(\Gamma_i)$ are adjacent, so are $\psi(v_{l+1}), \psi(v'_{l+1})$ since φ and φ^{-1} are isomorphisms. Thus $\psi(v_1)\cdots\psi(v_{l+1})$ and $\psi(v_1)\cdots\psi(v'_{l+1})$ are adjacent.

Injectivity and surjectivity of α again follow from the fact that φ and φ^{-1} are isomorphisms. □

Lemma 3.1.2. Let $\Gamma_1, \ldots, \Gamma_n$ be connected graphs and suppose that there exists an isomorphism $\varphi : \Gamma_1 \longrightarrow \Gamma_2$. Then if $\mathcal{S}$ and $\mathcal{S}'$ are Γ_1 and Γ_2-sheets respectively attached at the vertex $\tilde{v} \in V(\ast_{i=1}^n \Gamma_i)$ with $\varphi(u_1(\tilde{v})) = u_2(\tilde{v})$, then there exists an automorphism $\alpha \in \mathrm{Aut}(\ast_{i=1}^n \Gamma_i)$ such that $\alpha(\mathcal{S}) = \mathcal{S}'$, $\alpha(\mathcal{S}') = \mathcal{S}$ and α fixes the vertex $\tilde{v}$.

Proof: Let $\Gamma = \ast_{i=1}^n \Gamma_i$ be the free product graph described in the lemma with the sheets $\mathcal{S}$ and $\mathcal{S}'$ attached at $\tilde{v}$. Then create a new free product graph $\Gamma' = \ast_{i=1}^n \Gamma_i$ whose update function is initialised as $\mathbf{u}(\tilde{v})$. Then by Proposition 3.8 in [1], there exists an isomorphism $\mu : \Gamma \longrightarrow \Gamma'$ such that $\mu(\tilde{v}) = \varnothing'$, where $\varnothing'$ is the string of length zero in Γ', and each of the sheets attached at $\tilde{v} \in V(\Gamma)$ are mapped to their corresponding sheets at $\varnothing' \in V(\Gamma')$. Then the required isomorphism is $\alpha' = \mu^{-1} \circ \alpha \circ \mu$ where α is the automorphisms defined in the previous lemma. □

Theorem 3.1.3. Let $G \leqslant \mathrm{Aut}(\ast_{i=1}^n \Gamma_i)$ be the subgroup of the automorphism group of the free product graph only containing automorphisms that map sheets to sheets. If $\alpha \in G$, then α satisfies one of the following conditions:

- $\|\alpha\| = 0$ and α stabilises a sheet in $\ast_{i=1}^n \Gamma_i$
- $\|\alpha\| = 1$, α does not stabilise a sheet and α is the composition of an automorphisms α_v that fixes a vertex and an automorphism α_s that stabilises a sheet in the free product graph $\ast_{i=1}^n \Gamma_i$
- $\|\alpha\| > 1$ and α is a non-trivial translation along some bi-infinite string of sheets in $\ast_{i=1}^n \Gamma_i$

Proof: Suppose that $\alpha \in G$. Set $\min(\alpha) = \{\mathcal{S} \in \ast_{i=1}^n \Gamma_i : \mathfrak{d}(\mathcal{S}, \alpha(\mathcal{S})) = \|\alpha\|\}$. If $\|\alpha\| = 0$, then it is clear that α is stabilising a sheet in the free product graph by looking at the definitions of the functions $\|\cdot\|$ and $\mathfrak{d}$.

Now suppose that $\|\alpha\| = 1$ and $\mathcal{S} \in \min(\alpha)$. If α fixes a vertex than we can take α_s to be the identity automorphism. So suppose that α does not fix a vertex. Since $\|\alpha\| = 1$ we must have that α maps $\mathcal{S}$ onto an adjacent sheet $\mathcal{S}'$ i.e. $\alpha(\mathcal{S}) = \mathcal{S}'$ and keeps no sheet stabilised. First consider the case where we have the free product of only two graphs. Take the vertex $v \in V(\mathcal{S}) \cap V(\mathcal{S}')$ and consider the vertices $\alpha(v)$ and $\alpha^{-1}(v)$ in $*_{i=1}^{n}\Gamma_i$.

The sheet attached to $\mathcal{S}$ at $\alpha^{-1}(v)$, call it $\mathcal{S}''$, is mapped onto $\mathcal{S}$ under α (i.e. $\alpha(\mathcal{S}'') = \mathcal{S}$) and hence there must be an isomorphism between $\mathcal{S}$ and $\mathcal{S}'$ that maps the vertices in $\mathcal{S}$ and $\mathcal{S}'$ attached at v to each other. This follows since $\mathcal{S}'$ and $\mathcal{S}''$ are a copy of the same graph attached by the same vertex (due to the update function) to the sheet $\mathcal{S}$.

Then by Lemma 3.1.2, there exists an automorphism $\beta \in \mathrm{Aut}(*_{i=1}^{n}\Gamma_i)$ such that β fixes the vertex v and β swaps the sheets $\mathcal{S}$ and $\mathcal{S}'$. Then $\alpha \circ \beta$ is an automorphism of the free product graph that stabilises the sheet $\mathcal{S}$ and $\alpha = (\alpha \circ \beta) \circ \beta^{-1}$ is the composition of an automorphism that fixes a vertex and an automorphism that stabilises a sheet.

In the case where we are taking the free product of more than two graphs, there has to be an isomorphism between the sheets attached at $\alpha^{-1}(v)$ and the sheets attached at v. A similar argument can then be employed to construct the automorphism β.

If $\|\alpha\| > 1$ and $\mathcal{S} \in \min(\alpha)$, then the unique string of sheets containing the set of sheets $\{\alpha^m(\mathcal{S}) \mid m \in \mathbb{Z}\}$ forms a bi-infinite string of sheets that α translates along. □

Classifying the automorphisms of a free product graph in this way seems very natural and is similar in a sense to the way Tits classifies automorphisms of a tree. This classification makes a lot of sense especially when working with definition 2 as we can imagine the automorphisms of the underlying tree being automorphisms of the free product graph. It should be possible to extend this idea of an underlying tree of a free product graph to free products of more than two graphs and make inferences of the automorphisms of the free product graph by automorphisms of the underlying tree. This idea however is only possible in this case where we are only considering automorphisms that map sheets to sheets.

The reason behind having to restrict the automorphism group to a certain subgroup of automorphisms that only map sheets onto other sheets is because there exists particular free product graphs that have automorphisms mapping a single sheet onto pieces of multiple sheets. As an example, take the Cayley graph of the integers with respect to the standard generating set. This Cayley graph is a bi-infinite line and the free product of it with itself forms an infinite tree isomorphic to the Cayley graph of the free group on two generators (see Appendix 1 for a picture). Take the sheet that is the long (bi-infinite) horizontal line passing through the origin. There is an automorphism of this sheet that maps it onto any other bi-infinite path passing through the origin in the free product graph. This path could consist of possibly an infinite number of different sheets and this constitutes an automorphism that does not map a sheet onto a single other sheet. More work still needs to be done to provide a full classification of the automorphisms of a free product graph, however, the work done in this section provides a neat classification of the automorphisms contained in this particular subgroup of automorphism that map sheets to sheets. One question now would be to work out what types of free product graphs have their full automorphism group coincide with the subgroup G.

4 Conclusion

In this project, we have successfully proved that the four definitions of free products of graphs, those mentioned earlier in the piece, produce isomorphic graphs given that the input graphs are vertex-transitive. This shows that the free product of any Cayley graphs will be isomorphic no matter which definition of free product we choose, however, there is still some ambiguity when constructing the free product of graphs that are not vertex-transitive.

We have also developed a classification of certain automorphisms of a free product graph analogous to the way Jacque Tits classifies the automorphisms of a regular tree. This work further develops our understanding of automorphisms of free product graphs which will potentially lead to new outcomes in the theory of totally disconnected, locally compact groups. There is still, however, more work to be completed to provide a full classification of automorphism of a free product graph.

References

[1] Brawn, B., Robertson, D., Tornier, S., Willis, G., Free Products of Graphs, *Unpublished Manuscript*

[2] Möller, R.G., Seifter, N., Woess, W., Zemljič, S.S., Amalgamated free products of graphs and arc-types, *Unpublished Manuscript*

[3] Pisanski, T., Tucker T.W., 2002, Growth in products of graphs, *Australasian Journal of Combinatorics*, **26**, 155-169.

[4] Quenell, G., 1994, Combinatorics of Free Product Graphs, *Contemp. Math*, 257-281

[5] Tits, J., 1970, Sur le groupe des automorphismes d'un arbre, *Essays on Topology and Related Topics*, Springer-Verlag., 188-211

[6] Znöko, D.V., 1975, Free products of nets and free symmetrizers of graphs, *Math. USSR Sb.*, **27**, 463-480

[7] Gilch, L.A., Muller, S., 1975, Counting self-avoiding walks on free products of graphs, *Discrete Mathematics*, **340**, 325-332

[8] Meier, J., 2008, *Groups, Graphs and Trees: An Introduction to the Geometry of Infinite Groups*, Cambridge Univeristy Press

[9] Caprace, P., Monod, N. (Eds.), 2018, *New Directions in Locally Compact Groups* (London Mathematical Society Lecture Note Series), Cambridge University Press

[10] Le Boudec, A., 2016, *Groups of Automorphisms and Almost Automorphisms of Trees: Subgroups and Dynamics*, Lecture notes from Winter of Disconnectedness

Appendix 1: General Concepts & Terminology

Graph Theory

Throughout this paper the reader will encounter a substantial amount of graph theory terminology and ideas. This section aims to refresh the readers memory of typical graph theory concepts whilst setting out the notation that will be used throughout the rest of the paper. We will start out with some typical definitions that the reader should already be familiar with.

Definition (Graph). A *graph* $\Gamma = (V(\Gamma), E(\Gamma))$ is a pair of sets $V(\Gamma)$ and $E(\Gamma)$ where $E(\Gamma)$ is a collection of unordered pairs of elements from $V(\Gamma)$. We call the elements of $V(\Gamma)$ the vertices, the elements of $E(\Gamma)$ the edges and denote an edge between $v_1, v_2 \in V(\Gamma)$ by $\{v_1, v_2\}$. In the case when we consider the edge set $E(\Gamma)$ to consist of ordered pairs of vertices instead of unordered pairs, we will call the graph Γ a *directed graph* and the elements of E arcs. $\diamond$

The reader will probably be familiar with the geometric representation of a graph. A graph can be drawn in the plane with the vertices representing points in the plane and a line is drawn between two vertices $v_1, v_2 \in V(\Gamma)$ if $\{v_1, v_2\} \in E(\Gamma)$. In the case where Γ is a directed graph, an edge $(v_1, v_2) \in E(\Gamma)$ will be represented by an arrow from v_1 to v_2.

A vertex $v \in V(\Gamma)$ has *degree* n, denoted $deg(v)$, if there are n edges incident with v (or connecting to v). A graph is said to be k-regular or regular with degree k if every vertex in the graph has degree k.

Definition (Path). A *path* in a graph $\Gamma = (V(\Gamma), E(\Gamma))$ is a sequence $v_1, e_1, v_2, \ldots, v_{n-1}, e_{n-1}, v_n$ where $v_1, \ldots, v_n \in V(\Gamma)$, $e_1, \ldots, e_n \in E(\Gamma)$ and the edge e_i must connect the two vertices v_i and v_{i+1} for all $i \in \{1, \ldots, n-1\}$. We say that the path is *closed* if $v_1 = v_n$ and call the closed path a *cycle* if it has no repeated vertices. $\diamond$

A graph Γ is said to be *connected* if given any two vertices from Γ, we can find a path connecting the two vertices. A graph is called *bipartite* if its vertex set can be partitioned into two sets with no vertex in one set of the partition being adjacent to another member of that partition.

Later in this paper we will come across biregular trees. A *tree* is just the usual notion of a tree; a connected graph with no cycles or equivalently a connected graph of order n containing $n-1$ edges.

A Biregular tree is a tree that forms a bipartitite graph and any two vertices in the same set of the partition have the same degree. These trees are infinite as one would expect.

Definition (Graph Homomorphism). Let Γ_1 and Γ_2 be graphs. A homomorphism from Γ_1 to Γ_2 is a map $\psi : V(\Gamma_1) \to V(\Gamma_2)$ such that if $x_1, x_2 \in V(\Gamma_1)$ are adjacent, $\psi(x_1)$ and $\psi(x_2)$ must be adjacent in Γ_2. If ψ is injective then we call it an isomorphism and an isomorphism from a graph to itself is called an automorphism. $\diamond$

In this paper, we encounter groups of automorphisms acting as symmetries on a graph. This forms an important part of our study.

Free Groups & Free Products of Groups

Before we introduce the notion of a free group, we need to define what a 'word' is. Given a set of generators S for some group, a *word* in S is a sequence of elements from S. We call S the *alphabet* and say that a word is *reduced* if there is no element in the sequence adjacent to its formal inverse.

As an example, given the set $S = \{a, b\}$, $a, b, abab, a^3b^2$ are all words made up from the elements in S. If we now consider the set $S \cup S^{-1}$ as our alphabet, the word $abaa^{-1}b$ is not reduced and the reduced form of this word is ab^2.

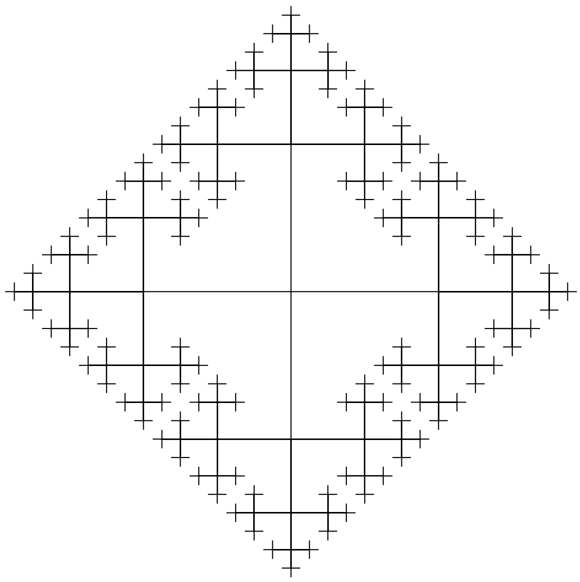
A free group is a group with generating set S and no word made up of the elements of S forms the identity element of the group apart from the identity element itself. Equivalently, a free group is a group whose presentation is of the form $G = \langle S, \varnothing \rangle$. We say that a free group is of *rank* n if its generating set has cardinality n. Every free group is made up of equivalence classes of words, where two words are equivalent if one can be reduced to form the other word. The free group of rank n is unique upto isomorphism and will be denoted by $\mathbb{F}_n$.

Another important concept that we discuss in this report is the notion of a free-product of groups. Given two groups G_1 and G_2, the free product of G_1 and G_2 denoted $G_1 * G_2$, is the group containing equivalence classes of all words formed from the elements of the two original groups and two words are considered equivalent if one can be reduced to the other. The group $G_1 * G_2$ is always infinite even if the original groups are finite (except for the case when one of the groups is trivial). Despite its name, the free product of two groups is not a free group unless the two original groups are free. However, the group $G_1 * G_2$ is the 'freest' group containing both G_1 and G_2 as subgroups. We will see later that there is a duality between free products of groups and free products of graphs.

Cayley Graphs

Let G be a groups with presentation $G = \langle S|R\rangle$ where S is the set of generators for the group G and R is the set of relations as discussed earlier. The Cayley graph of the G with respect to the generating set S, denoted $\Gamma_S(G)$, is the graph with vertex set G and two vertices $g_1, g_2 \in G$ have a directed edge between them (from g_1 to g_2) if there exists some $s \in S$ such that $g_1 s = g_2$. However, in this paper, we assume that the generating set S for the Cayley graph is closed under inverses and does not contain the identity. In this case where the generating set is closed under inverses, we can drop off the directions on the edges.

A simple example is the Cayley graph of the group $\mathbb{Z}_n$ which forms the cyclic graph on n vertices denoted C_n with respect to the generating set $\{1\}$. A more interesting example is the Cayley graph of the free group $\mathbb{F}_2$ which forms an infinite tree as pictured below:



Cayley graph of the free group $\mathbb{F}_2$