On Picard Groups and Jacobians of Directed Graphs

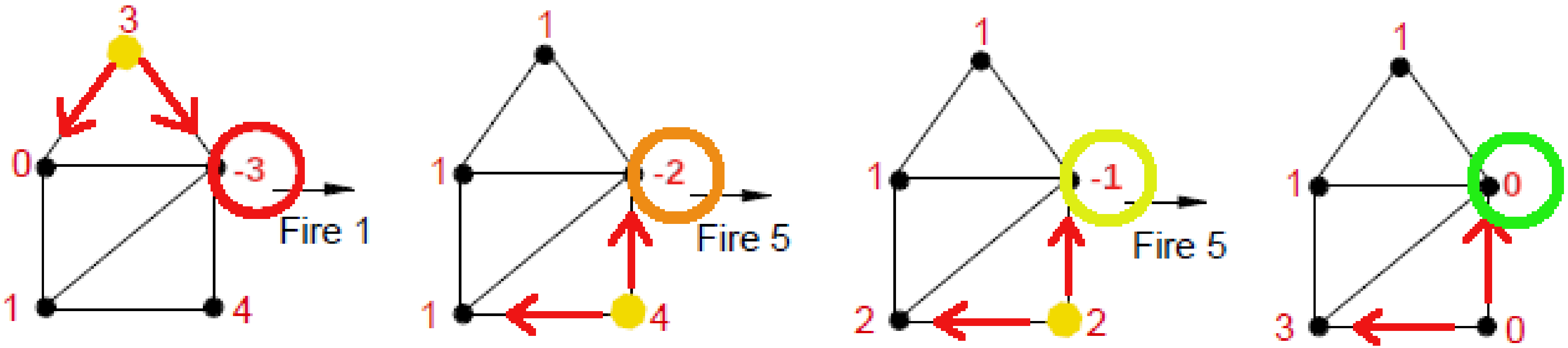
JAIUNG JUN
YOUNGSU KIM
MATTHEW PISANO

Introduction

- A chip firing game is a one player game played on a graph where integer amounts of chips are fired between vertices along their adjacent edges.
- Since its origin in 1983, it has since become an important tool in structural combinatorics and other areas of mathematics. For instance, from a perspective motivated by algebraic geometry, one may view finite graphs as a discrete model for Riemann surfaces.
- The main area of study of this game is on its standard, undirected variant. However, this game can also be played on directed graphs.
  - Though this area has been comparatively less researched, we hope to change that through the patterns and techniques we have developed in our investigation.

Preliminaries

The Chip Firing Game



The Chip Firing Game

- When a game is started, each vertex on a graph is assigned a certain number of chips.
- During play, chips can be lent or borrowed at each node where one or more chips are either sent or received along each outgoing edge equally.
- In the case of a directed graph, vertices can only interact with another along an outgoing or bidirectional edge.
- The game is won once every vertex has a zero or greater number of chips, meaning that no vertex is in debt.

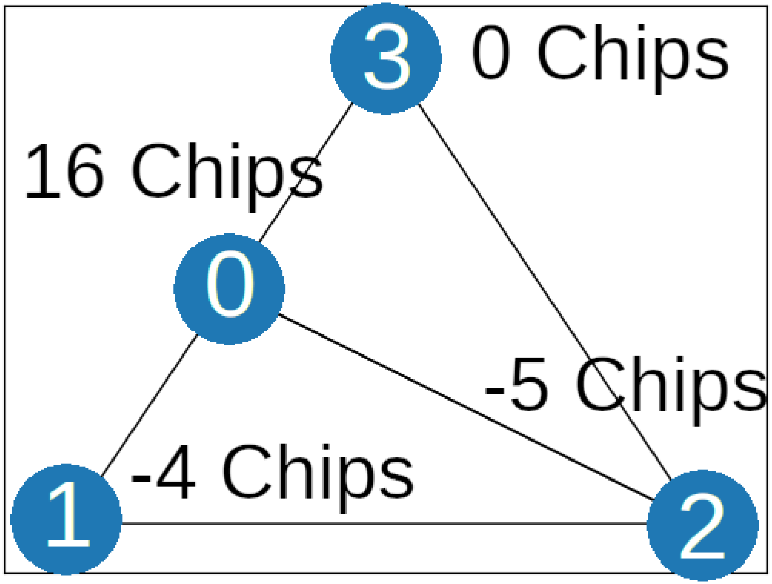
Applications

- By further understanding how to analyze the possible moves and winning strategies of more complex graphs, Chip-Firing games become more easily usable in different applications.
- Aside from their usage in mathematics, one notable usage of these games is in economics where these games, especially the directed variants, can be used to model the flow of money or assets from one entity to another.
  - A more entertaining usage is through map-based board games such as *RISK* where armies are sent to neutralize neighboring enemies until the game is won when no other enemies are on the board.

Terminal Strong Components

- Terminal strong components are sub-graphs of a larger oriented graph.
  - They are defined as regions of a graph that are strongly connected, in other words, every vertex has a directed path to all other vertices.
    - For undirected graphs, all sub-graphs are strongly connected.
  - They are also terminal, as in there are only edges pointing into the sub-graph, with none coming out of it.
- Understanding these regions is important for understanding how the game is played. An example of their importance is how chips can only be lent into these regions, trapping them until a borrowing move has been made.

Divisors

G = 

Where divisor

D = [16, -4, -5,0]

Divisors and Equivalence Relations

- In the study of this game a **Divisor** of a graph *G*, is an integer vector $v \in \mathbb{Z}^n$ where *n* is the number of vertices in the graph. The i^{th} element of *v* is the number of chips on the i^{th} vertex of the graph.
- Two divisors have an **Equivalence Relation** (~) if one divisor can be obtained from the other by a finite series of lending or borrowing moves made on a certain graph.
- An ***Equivalence Class***, *[D]*, is the set of all divisors that are equivalent to each other when based on that graph.
- The collection of all divisors on a graph defines a free abelian group Div(G), the divisor group of G. All members of this group are related to one of the graph's equivalence classes and are generated through some combination of lending or borrowing moves.

The Picard Group and The Jacobian

- The **Picard Group** of a graph, *Pic(G)*, is the set of all equivalence classes that the divisors of that graph *G* can be a part of. The larger the size of the Picard Group, the more ways a game can be played.
  - The Picard Group is a finitely generated abelian group. All of its member vectors can be added or subtracted and still be within that group.
- The degree of a divisor or an equivalence class is the sum of each of the divisor's elements.
- The **Jacobian** of a graph, *Jac(G)*, is the torsion sub-group, a special subset, of *Pic(G)* such that every divisor in each equivalence class has a degree of *0.*
  - All of its members can be generated by some combination of divisors within its equivalence classes.
  - If a divisor is in one of the Jacobian's classes, it can be made winning after a finite series of moves. The larger the size of the Jacobian, the more configurations exist where the vertices can be made debt free.

The Picard Group and The Jacobian

The Picard Group is comprised of two parts:

- The Jacobian
  - The Jacobian itself is comprised of one or more invariant factors which are each in the form of $\mathbb{Z}_x$. Here, *x* represents the number of distinct equivalence classes with a degree of zero the graph can support.
  - Two invariant factors multiplied together represent a tuple. These represent a graph state that is a combination of multiple classes combined together. This is similar to a basis would function in a vector space.
- The sets of integers to the power of its rank
  - In the form of $\mathbb{Z}^n$ *(an n-tuple of integers)* where *n* is the rank, representing the number of ways any number of chips can be distributed along classes represented by the Jacobian. This tuple can represent the scaling up or down of the invariant factor(s) that make up the Jacobian

Using the Picard Group, we can completely describe any initial or intermediate state of a game, given the graph that it is played on.

The Laplacian and The Smith Normal Form

- The **Laplacian** of a graph helps to serve as a bridge between the conceptual game and the mathematics behind those concepts.
  - For a graph of size *n*, it is an *n x n* matrix representing all valid lending or borrowing moves that graph can make. Multiplying the transpose of the i*th* row of the Laplacian by a divisor results in the divisor of the graph after making a move at the i*th* vertex.
- The **Smith Normal Form** (SNF) of a Laplacian is a matrix obtained from the Laplacian. While the Laplacian itself encodes information about lending or borrowing moves, the SNF encodes information about the Picard Group and the Jacobian in its diagonal elements.
  - Calculating the SNF allows us to know more information on the possible ways a game can be played out. Similarly to Gaussian elimination, the Laplacian is reduced to a diagonal matrix through a series of row and column operations. The difference being that all elements of the matrix must be integers.

The Laplacian and The Smith Normal Form

Laplacian

	0	1	2	3	4	5	6	7
0	2.00000	-1.00000	-0.00000	-0.00000	-1.00000	-0.00000	-0.00000	-0.00000
1	-1.00000	2.00000	-0.00000	-1.00000	-0.00000	-0.00000	-0.00000	-0.00000
2	-0.00000	-0.00000	1.00000	-1.00000	-0.00000	-0.00000	-0.00000	-0.00000
3	-0.00000	-0.00000	-0.00000	0.00000	-0.00000	-0.00000	-0.00000	-0.00000
4	-0.00000	-0.00000	-1.00000	-0.00000	1.00000	-0.00000	-0.00000	-0.00000
5	-0.00000	-0.00000	-0.00000	-0.00000	-1.00000	3.00000	-1.00000	-1.00000
6	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	0.00000	-0.00000
7	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	0.00000

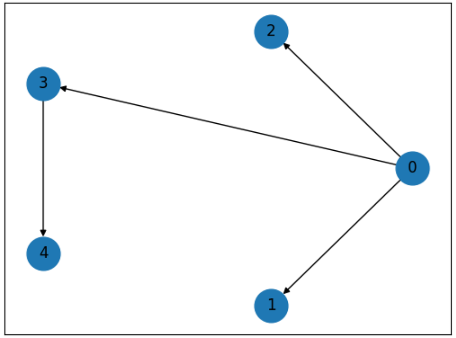
Smith Normal Form

	0	1	2	3	4	5	6	7
0	1.00000	-0.00000	0.00000	0.00000	0.00000	-0.00000	-0.00000	0.00000
1	0.00000	1.00000	-0.00000	0.00000	0.00000	0.00000	-0.00000	0.00000
2	0.00000	-0.00000	1.00000	0.00000	0.00000	0.00000	-0.00000	0.00000
3	-0.00000	0.00000	0.00000	1.00000	0.00000	-0.00000	0.00000	0.00000
4	0.00000	-0.00000	-0.00000	-0.00000	3.00000	0.00000	-0.00000	-0.00000
5	-0.00000	0.00000	0.00000	-0.00000	0.00000	-0.00000	0.00000	0.00000
6	-0.00000	0.00000	0.00000	-0.00000	-0.00000	-0.00000	0.00000	0.00000
7	0.00000	-0.00000	-0.00000	-0.00000	0.00000	0.00000	-0.00000	0.00000

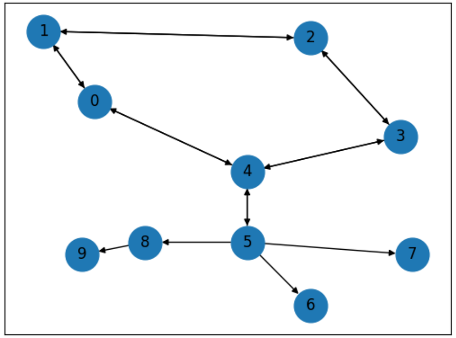
The non-zero elements of the diagonal represent the Jacobian of a graph while the diagonal zeros represent the rank of the Picard Group. Here, $Pic(G) = \mathbb{Z}_3 \times \mathbb{Z}^3$, the Jacobian comes from the 3 at $M_{4,4}$ and the rank from 3 empty rows.

Focused Graphs

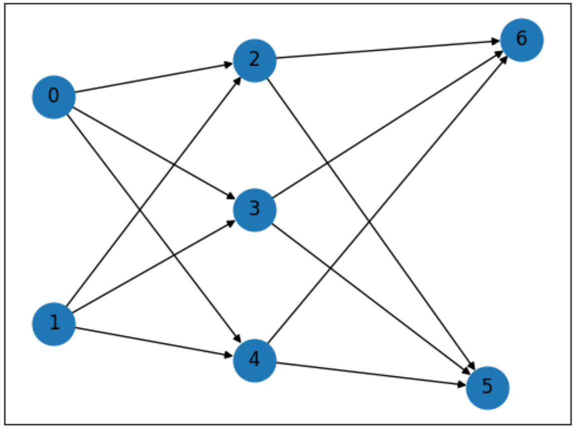
Tree Graph



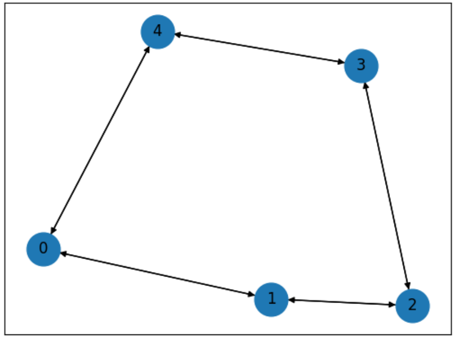
Pseudo-Tree Graph



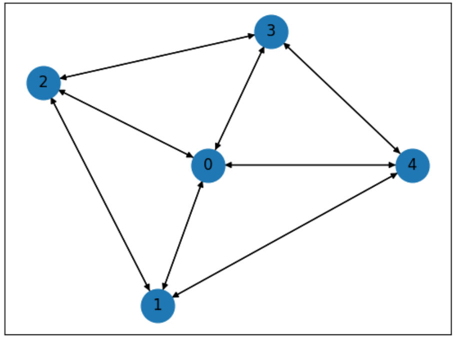
Multipartite Graph



Cycle Graph



Wheel Graph



Focused Graphs

- A **Tree Graph** is a graph where there is only one path between vertices and contains no cycles.
- A **Cycle Graph** is a graph that only has one cycle, or a line graph with another connection between the first and last vertices.
- A **Pseudo-Tree Graph** is a combination of these two. This graph is created by gluing a tree to one of the vertices of a cycle graph.
- A **Wheel Graph** is a cycle graph with an added central vertex to which all others connect.
- A **Multipartite Graph** is a graph made up of several groups of vertices in which their vertices have no connection to each other, but are each directionally connected to the next such group.

Research

Objectives

- While chip firing games with undirected graphs are well studied and explored, the directed case has not received as much attention.
- Our goal is to explore ways to calculate these directed graphs and to study their relationships with their undirected counterparts.

Methods

- By using our focused graphs as a guide, we conducted our research by looking for patterns within different configurations and graph sizes.
- We then computed many examples of said configuration to see if our original guesses held up.
- If that was the case, we then moved on to rigorously proving the conjectures that we could and adding even more computational results to those that we could not.

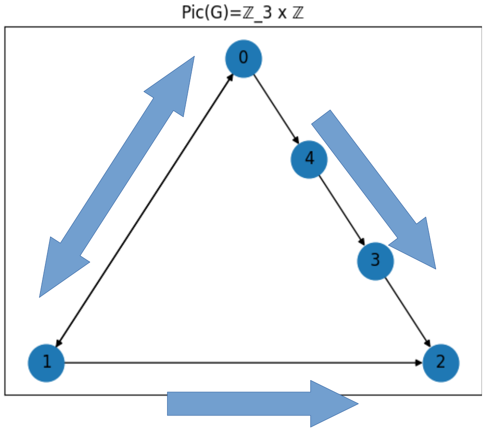
Finding a Tree's Picard Group

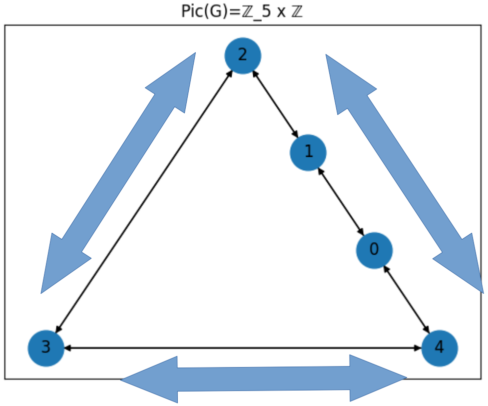
- The Picard group is commonly written in the form $Pic(G) = Jac(G) \times \mathbb{Z}^n$, where n is the rank of the Picard group.
  - We have noticed that this rank for a tree graph can be easily calculated inductively. By reconstructing an arbitrary tree edge by edge, its rank can be determined by following two rules *(thrm)*.
    - If the next arrow drawn is pointing towards the graph or if it is bidirectional, the rank does not change.
    - If the next arrow is pointing towards the new vertex, the rank increases by one so long as the number of terminal strong components also grows as a result.
  - We can see that the rank of a tree corresponds to the number of terminal strong components of that tree, sections that are only connected to the rest of the graph by an incoming edge and have a path between all of its member vertices.
- The Jacobian of a tree is relatively simple, it is always the trivial group *(thrm)*.
  - We have also proven this through induction as there are no cases in which it changes as more and more edges are added.

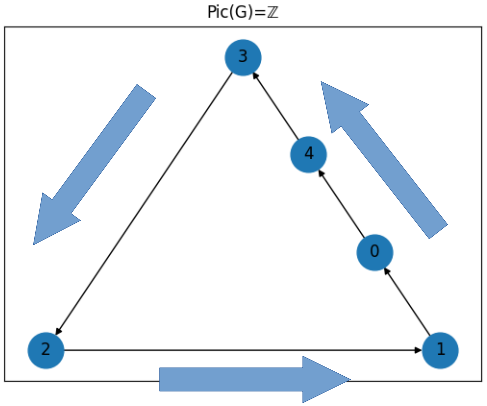
Creating a Pseudo-Tree

- A Pseudo-tree can be created by gluing a tree to a cycle graph in one of two ways.
- By Vertex – Here, whichever vertices will be glued together will be merged into one vertex. With this way of gluing, one vertex will be shared between the two glued graphs.
- By Edge – With this method, the two graphs are joined by an additional edge. This helps to preserve the attributes of the original graphs into the resulting glues pseudo-tree, such as the Jacobian often being *Jac(cycle) x Jac(tree)*.

Finding The Picard Group of a Cycle Graph





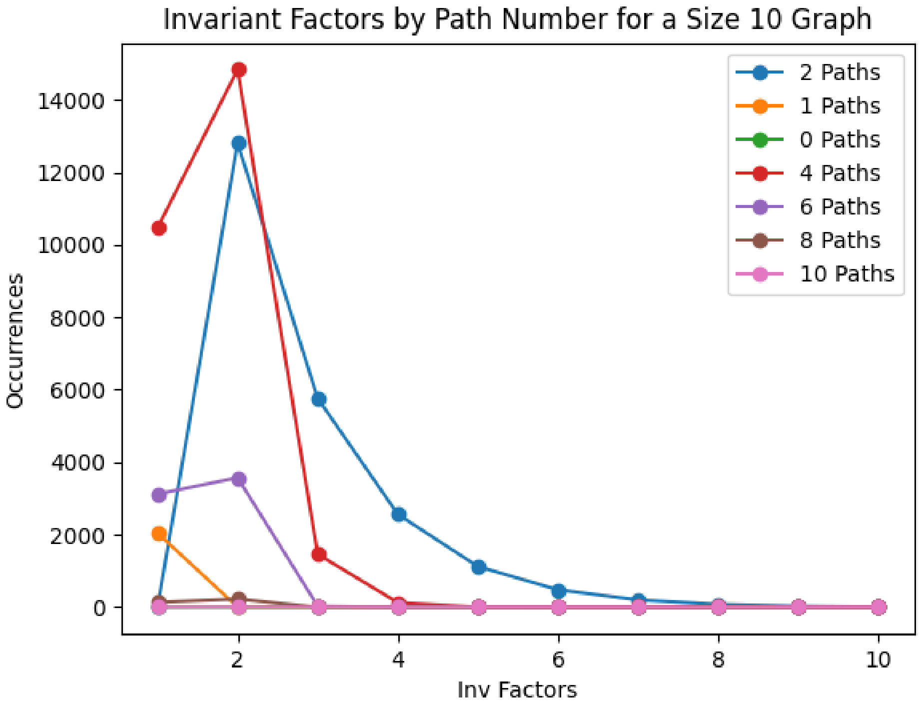


Finding The Picard Group of a Cycle Graph

- The rank of the Picard group of a cycle graph is similar to that of a tree. It is the number of terminal strong components in the graph.
- The Jacobian of a Cycle graph is more complex than that of a tree, we have proved that there is always some orientation of a cycle such that the Jacobian is trivial or $\mathbb{Z}_k$ where $k \leq n$ when $n \geq 3$ *(thrm)*.
  - The most common of these invariant factors are the trivial factor and $\mathbb{Z}_2$ across all possible orientations.
- Additionally, we have been able to calculate the Jacobian for any arbitrary cycle graph (C_n) with two *paths.* Here a *path* is a connected sub-graph of C_n in which all arrows are oriented in a single direction or are bidirectional. In other words, a path is a sub-graph of C_n with one strong terminal component. These paths either only comprise one edge or are terminated by one appropriate directed edge on each side *(thrm)*.
  - For these graphs, the Jacobian is $\mathbb{Z}_{(x+2)}$ where x is the number of bidirectional edges clockwise of the counter-clockwise path and counter-clockwise of the clockwise path.

Finding The Picard Group of a Cycle Graph

A demonstration of the distribution of invariant factors

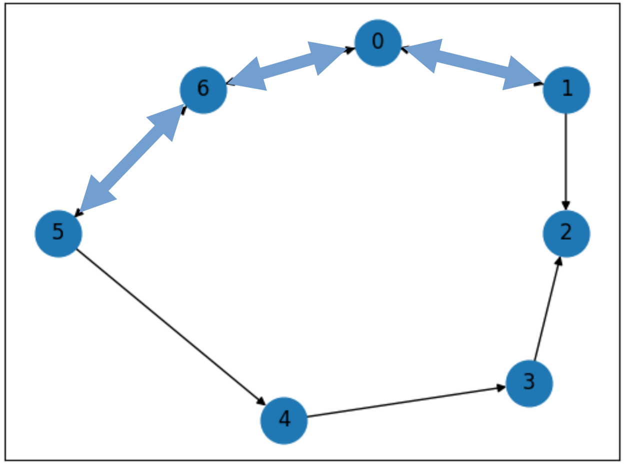


Finding The Picard Group of a Wheel Graph

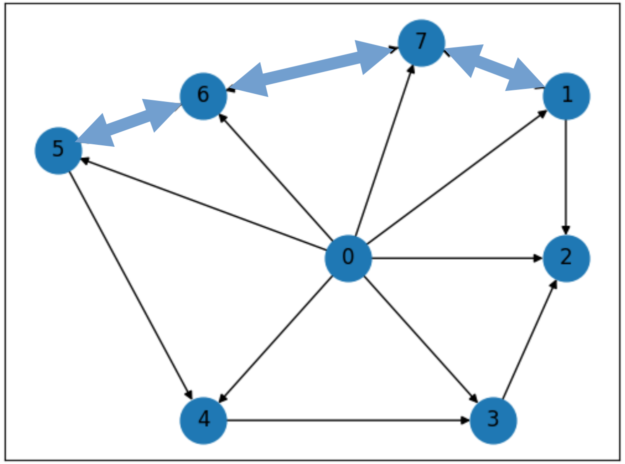
- For wheel graphs, we looked for patterns that arose within the invariant factors of the Jacobian as a general formula was not immediately obvious. For this strategy, we broke the edges of the wheel graph into their two most obvious groups, those belonging to the rim of the wheel and those of the spokes. By orienting all the edges of either group the same way and trying all nine combinations, we noticed a well-defined pattern for each as the size of the wheel graph changed. These patterns fell into four distinct cases.
- The most interesting of these cases was for graphs whose rims were bidirectional and whose spokes pointed inward. This orientation produces a result very similar to that of an undirected wheel graph where the Jacobians followed the pattern of $\mathbb{Z}_{\alpha\varphi\hat{}n} x\ \mathbb{Z}_{c\alpha\varphi\hat{}n}$ when the size was odd where $\alpha \cong 0.27555$ and $\mathbb{Z}_{\beta\varphi\hat{}n} x\ \mathbb{Z}_{\beta\varphi\hat{}n}$ when the size was even where $\beta \cong 0.618035$. In both of these patterns, φ represents the golden ratio *(conj)*.
  - In our directed case, $c=4$ while in the undirected case, $c=5$.
  - This can also be modeled as a series similar to the Lucas numbers.

Connections Between Wheel and Cycle Graphs

$\mathrm{Pic}(C_7) = \mathbb{Z}_5 \times \mathbb{Z}$



$\mathrm{Pic}(W_8) = \mathbb{Z}_{35} \times \mathbb{Z}$



Connections Between Wheel and Cycle Graphs

- During our experiments with wheel graphs whose spokes pointed outward, their Picard groups behave similarly to the cycle graph one size smaller, as if the central vertex was not there at all. This was true for any arbitrary orientation of the rim *(thrm)*.
- This is due to the fact that chips are only fired along outgoing edges. If all of the spokes point outward, the vertices on the rim cannot interact with the axle, functioning similarly to a cycle graph.
- These relations are also split into several cases, for example:
  - For prime cycle graphs and their counterparts, arbitrary Picard groups of wheel$_n$ appear to be $\mathbb{Z}_{(n-1)*a} \times \mathbb{Z}$ where a is in the Picard group of Cycle$_{(n-1)}$ $\mathbb{Z}_a \times \mathbb{Z}$ when all spokes point outward.

Finding The Picard Group of a Multipartite Graph

- The structure of these graphs that we investigate are intentionally designed to resemble artificial neural networks. To further facilitate this comparison, we direct all edges *forward* such that, after numbering the groupings of these vertices in some order, edges always point towards the next highest numbered grouping.
- We were able to find notable patterns in both a *Perceptron* style model with two layers and a *Hidden Layer* model with three layers *(conj)*.
  - For two layers in the form of $f \to s$ where f and s are the number of nodes in the first and second layers, respectively. For these graphs, $Pic(G) = \mathbb{Z}_{f-1}{}^{s} \times \mathbb{Z}^{s}$.
  - For the three layer model, things once again become more complex, being split into cases. There are similarly structured to the two layer models where the invariant factors are based off of the size of all three layers and the rank is just the size of the last layer.