NECKLACES AND SLIMES

by

Jina Park

A thesis submitted to the Graduate Council of
Texas State University in partial fulfillment
of the requirement for the degree of
Master of Science
with a Major in Mathematics
May 2020

Committee Members:

Suho Oh, Chair

Anton Dochtermann

Eugene Curtin

Jian Shen

FAIR USE AND AUTHOR'S PERMISSION STATEMENT

DEDICATION

To the Lord Jesus Christ who has blessed me to study mathematics,
and
to my family who have supported me with lavishing love and cheer.

ACKNOWLEDGEMENTS

Thank you to my advisor, Dr. Oh for his wonderful math instruction, guidance, assistance, mentorship, and patience throughout all the process of this study.

Thank you to my professors in the math department for their knowledge, discipline, and devotion that help me finish all math course work.

Thank you to the thesis committee for giving their precious time to review my thesis draft and help me revise it until the last moment.

Thank you to Dr. Welsh and Dr. White for helping me successfully change the program to a master's so that I can write this thesis.

Thank you to Ms. Amanda Garza for her persistent assistance, loving-kindness, and support throughout.

TABLE OF CONTENTS

LIST OF FIGURES

Figure **Page**

ABSTRACT

It was asked if one can find a bijective map between the following two objects: binary necklaces with n black beads and k white beads and certain (n, k)-codes whose weighted sum is 0 modulo n (Brauner et al.,2019 [9]; Chan, 2019 [10]). The former object is one that has been studied for ages, whereas the latter one was shown to be the states in a dollar game played on a cyclic graph (Corry & Perkinson, 2018 [11]).

The case when n and k are coprime was shown in [9] and it is easily described by using rotation. We show that in the general case, all that one needs to construct the bijective map is to construct a rotation-invariant and weight increasing map (riwi-map) on the codes. When n and k are coprime the simple cyclic rotation works as a riwi-map. We show that when n or k is prime, a new map called a slime migration works as a riwi-map and hence allows one to get a bijective map as a result.

I. INTRODUCTION

Review of Necklace Problems

In combinatorics, a q-ary necklace of length n is an equivalence class of words of length n on q symbols, taking all rotations as equivalent. Thus, two necklaces are inequivalent if they cannot be transformed from one to the other under the circular shift. According to the research by Berstel and Perrin [5], circular words or sequences are commonly produced in the process of analyzing periodic phenomena that occur in many fields such as music, astronomy, and biology. In combinatorics, circular words or necklaces have a wide range of applications and interpretations in algebra, geometry, and combinatorial enumeration.

The cardinality of q-ary necklaces of length n has been known for a long time. The formula for the number of q-ary necklaces of length n

$$M_q(n) = \frac{1}{n} \sum_{d|n} \varphi(d) q^{n/d}$$

where $\varphi(d)$ is Euler's totient function, is credited to C. Moreau (1872) (as cited in [5]).

Stanley [16] showed in the problem 105(a) of *Enumerative Combinatorics* (1986) that the number of binary necklaces of length n is

$$\frac{1}{n} \sum_{d|n} \varphi(d) 2^{n/d} = f(n)$$

where $f(n)$ denotes the number of subsets of $\mathbb{Z}/n\mathbb{Z}$ whose elements sum to 0 in $\mathbb{Z}/n\mathbb{Z}$, when n is odd.

For instance, the number of binary necklaces of length 5 is

$$\frac{1}{5} \sum_{d|5} \varphi(d) 2^{5/d} = \frac{1}{5}\{\varphi(1) \cdot 2^5 + \varphi(5) \cdot 2\} = \frac{1}{5}(32 + 4 \cdot 2) = 8.$$

We can see that this cardinality equals

$$f(5) = |\varnothing, \{0\}, \{1,4\}, \{2,3\}, \{0,1,4\}, \{0,2,3\}, \{1,4,2,3\}, \{0,1,2,3,4\}| = 8.$$

To give an example, we visualize $f(5)$ by matching it to the configurations of binary necklaces in Figure 1.1. Let us label the positions in a configuration from 0 to 4 starting from the topmost position and going around clockwise. Then, the elements of each subset of $\mathbb{Z}/n\mathbb{Z}$ whose elements sum to 0 (mod n) correspond to the positions for white beads to be placed. For instance, $\varnothing$ corresponds to the left-most configuration in the first row since there is no place for a white bead. By the same way, we can map $\{2,3\}$ to the right-most configuration in the first row and $\{0,1,4\}$ to the left-most configuration in the second row.

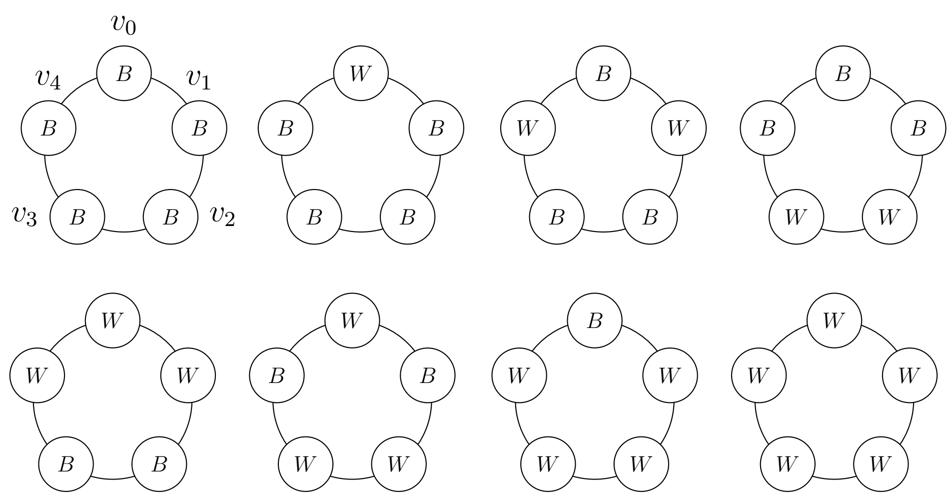

FIGURE 1.1: $f(5)$ configurations

Stanley [16] asked if there is a combinatorial proof for a bijection between the set of binary necklaces of length n and the set of subsets of $\mathbb{Z}/n\mathbb{Z}$ whose elements sum to 0 (mod n). Chan [10] gave an answer to this question by specializing to $q = 2$ by proving there is a bijection between the set of q-ary necklaces of length n and the set of codes with n entries from $\{0, ..., q-1\}$ such that their weighted sum equals 0 modulo n when n and q are coprime.

Also, Chan [10] asked if there is a bijection between the set of binary necklaces of length n with k black beads, and the set of functions $f : \mathbb{Z}/n\mathbb{Z} \mapsto \{0, 1\}$ such that their weighted sum is divisible by n, and the number of subsets of $\mathbb{Z}/n\mathbb{Z}$ whose elements sum to k. Thereafter, Brauner, et al., [9] defined the set $\mathcal{N}_{n,k}$ of binary necklaces with n black beads and k white beads and proved that in the case n and k are coprime there is a bijection between $\mathcal{N}_{n,k}$ and the set $\mathcal{F}_{n,k,0}$ of (n,k)-codes whose degree is k and weighted sum is divisible by n based on the divisor theory of graphs.

However, a bijection between these two sets in the general case is still being sought. We come up with one in the case n is prime via a so-called ***riwi***-map. In the particular case that n is an odd prime and not coprime to k, a ***slime migration*** works as a riwi-map. Briefly speaking, a slime migration occurs in an (n,k)-code where several ***slimes*** move in the same direction making the weighted sum increased. The slime migration is realized as a key factor that makes a bijection between $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$ viable under the condition n and k are not coprime. Therefore, our primary goal is to articulate the scheme for the construction of a slime migration and of a bijection determined by this migration. First, let us describe the main objects and terminology that are going be used throughout this thesis.

A New Bijection

The following sets are the main objects that we study here.

- The set $\mathcal{N}_{n,k}$ of ***binary necklaces*** (i.e. strings equivalent up to cyclic rotations) of length $n+k$ using n black beads and k white beads.

- The set $\mathcal{F}_{n,k}$ of (n,k)***-codes***, functions $f:[n]\to\mathbb{Z}^{\geq 0}$, for which their sum is k. The set $\mathcal{F}_{n,k}$ is further divided into sets $\mathcal{F}_{n,k,t}$ of (n,k,t)-codes for $t\in\{0,...,n-1\}$, where the weighted sum is t modulo n:

$$\mathcal{F}_{n,k,t}:=\Big\{f|\sum_{i\in[n]}f(i)=k,\ \sum_{i\in[n]}i\cdot f(i)\equiv t \pmod{n}\Big\}.$$

Research by Chan [10] showed that the cardinality of $\mathcal{N}_{n,k}$ is equal to the cardinality of $\mathcal{F}_{n,k,0}$ in the case n is an odd positive integer. The cardinality of $\mathcal{N}_{n,k}$ was computed by Ardila et al. [2] and the cardinality of $\mathcal{F}_{n,k,0}$ is obtained by a calculating method of Kitchloo and Parcher [13]; each cardinality equals

$$\frac{1}{n+k}\sum_{m|n,m|k}\varphi(m)\binom{(n+k)/m}{n/m}$$

where $\varphi(m) = \#\{p \in \mathbb{N} : 1 \leq p \leq m,\ \gcd(m,p) = 1\}$ is Euler's totient function.

A combinatorial bijection between $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$ is pretty easy to describe when n and k are coprime. A rotation map gives a nice bijection in the case that n and k are coprime. The weighted sum of an (n,k)-code increases by k upon rotations. Hence, if n and k are coprime, k generates $\mathbb{Z}/n\mathbb{Z}$. This induces a natural bijection between $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$ in the case n and k are coprime.

Take a look at the configurations of Figure 1.2 and give them codes $f_1,\ f_2,\ f_3,\ f_4$ and f_5, respectively. Then, under a rotation map c, $c^3(f_1) = f_4$, $c^2(f_2) = f_4$, $c^1(f_3) = f_4$, $c^0(f_4) = f_4$, and $c^4(f_5) = f_4$ where $f_4 \in \mathcal{F}_{5,3,0}$. This demonstrates that any code representing a necklace of $\mathcal{N}_{5,3}$ can be sent to a code in $\mathcal{F}_{5,3,0}$ by rotating several times.

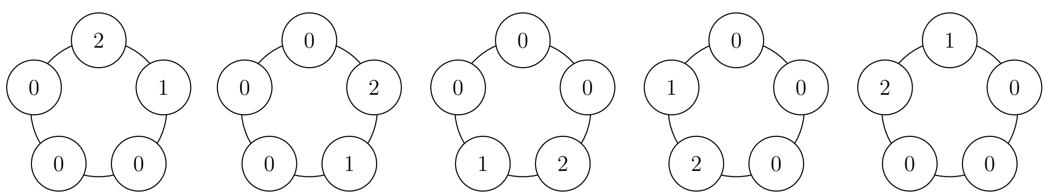

FIGURE 1.2: A binary necklace of $\mathcal{N}_{5,3}$

On the other hand, Figure 1.3 clearly shows that n does not need to be prime as long as it is coprime to k. Rotating three times of a code f given to the first configuration of $\mathcal{N}_{4,5}$ sends it to the code $\mathcal{F}_{4,5,0}$ representing the third configuration.

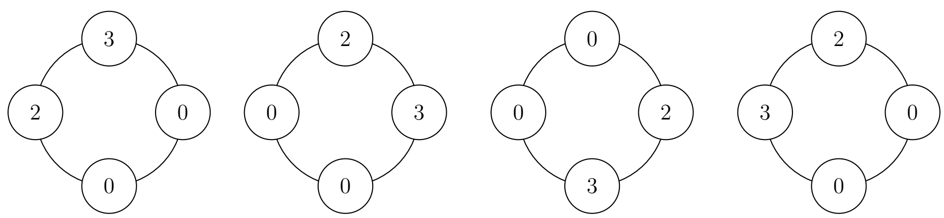

FIGURE 1.3: A binary necklace of $\mathcal{N}_{4,5}$

Unfortunately, this approach does not work when n and k are not coprime since k does not generate $\mathbb{Z}/n\mathbb{Z}$ in this case. Figure 1.4 illustrates this through $\mathcal{N}_{4,6}$ necklaces. Let f be a code representing the leftmost configuration of Figure 1.4. Then, $f \in \mathcal{F}_{4,6,3}$, $c(f) \in \mathcal{F}_{4,6,1}$, $c^2(f) \in \mathcal{F}_{4,6,3}$, and $c^3(f) \in \mathcal{F}_{4,6,1}$. The weighted sum increases by 2 upon rotations but none of them belongs to a set $\mathcal{F}_{4,6,0}$.

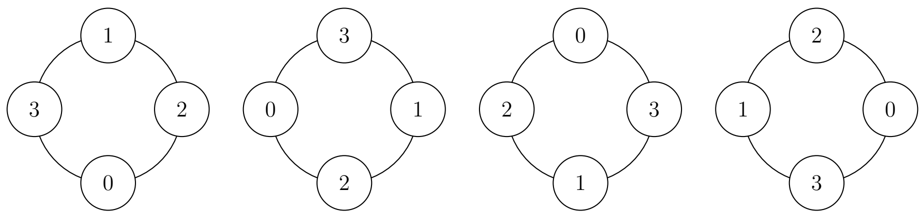

FIGURE 1.4: A binary necklace of $\mathcal{N}_{4,6}$

For another instance, let us consider a necklace of $\mathcal{N}_{5,10}$ in Figure 1.5. Give a code f to the leftmost configuration of Figure 1.5, then $f \in \mathcal{F}_{5,10,1}$. Upon rotations, the weighted sum of f increases by $k = 10 \equiv 0$ modulo 5. Thus, all codes obtained from f by rotations belong to the same set $\mathcal{F}_{5,10,1}$. We see that whether n is prime does not matter for a bijection between $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$.

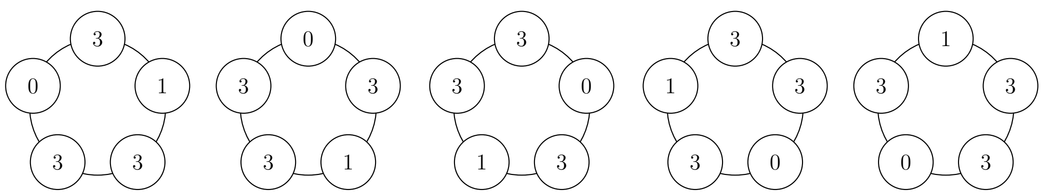

FIGURE 1.5: A binary necklace of $\mathcal{N}_{5,10}$

Therefore, one approach would be to construct a new map that is different from rotation, but still provides a nice bijection between $\mathcal{F}_{n,k,t}$'s regardless whether n is coprime to k. To construct such a bijective map, we lay down a two-step scheme. The first step is to construct a bijective map that is rotation invariant and sends a code in $\mathcal{F}_{n,k,t}$ to code in $\mathcal{F}_{n,k,t+1}$ (excluding one object from $\mathcal{F}_{n,k,0}$); that is a riwi-map. If such a map is realized, we can build up a bijection between $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$ by applying it enough times. For this reason, a rotation map can work as a riwi-map in the case n and k are coprime. The slime migration that we construct successfully meets these conditions so that it can take a role as a riwi-map when n is an odd prime and not coprime to k. Upon a slime migration, the weighted sum of an (n, k)-code increases by a certain number that is coprime to n. Therefore, a slime migration effectively works as a bijection. The next step is to systematize a new bijection between $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$ based on such a riwi-map. In this process, a notable phenomenon is found. A bijection realized by a slime migration sends a set of codes in $\mathcal{F}_{n,k,0}$ to a set of (n, k)-codes given to the different necklaces of $\mathcal{N}_{n,k}$ whereas a bijection established by a rotation map sends a single code in $\mathcal{F}_{n,k,0}$ to code representing a necklace of in $\mathcal{N}_{n,k}$.

Recall that the slime migration occurs in an (n, k)-code, which generally means a code given to a necklace of $\mathcal{N}_{n,k}$. Thus, the slime migration seems to be more related to $\mathcal{N}_{n,k}$ than $\mathcal{F}_{n,k,0}$. For this reason, it is suitable to hold up the set $\mathcal{F}_{n,k,0}$ before going further to a slime migration on an (n, k)-code and a bijection between $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$. We found that there is a bijection between $\mathcal{F}_{n,k,0}$ and a set of out-of-debt chip-firing states while going through the review of a chip-firing game and a dollar game. Furthermore, this review facilitates a multilayered understanding of and insight to the weighted sum, linearly equivalent divisors, and affiliated graphs.

Out-of-Debt Chip-Firing States

A brief history of a chip-firing game

According to the overview by Biggs [6], a chip-firing game on a graph G is run by a sequence of firing chips at the vertices where a pile of chips is placed. A vertex v fires when the number of chips on it is at least its degree. This action is called a ***legal move***. By a legal set-move, one chip is sent from v to each adjacent vertex. The process of firing is continued until a ***stable*** state is reached where a stable state means one for which no further legal firing is available at every vertex of G.

Klivans [14] provides a brief overview of the origin of a chip-firing game as a combinatorial game. The following is mainly based on her research. Spencer [15] began to study a balancing game from a combinatorial point of view. He initially placed chips at the origin of $\mathbb{R}$ and considered to move the same number of chips to its neighboring sites. It is said that Spencer's motivation was to "balance a collection of vectors in the max norm" (as cited in [14], p.9). Following this work, Anderson et al., [1] investigated the diverse properties of firing chips from the origin, the final stable configurations, and the exact number of fires required at each site. Thereafter, properties of chip-firing on arbitrary undirected or directed graphs, consideration of the arbitrary initial configurations, and fundamentals of the finite and infinite process according to the initial number of chips have been broadly explored by Björner and Loász [7] and Björner et al. [8]. Biggs [6] came up with the concept of the group of ***critical*** configuration of a graph. A configuration is critical when it is both stable and recurrent. A configuration r is ***recurrent*** when the configurations generated from r through a series of firing at vertices reach r. The idea of this group is connected to the lattices of cuts and flows of the graph by Bacher et al. [3].

A dollar game

The legal moves in a dollar game stick to the same rules of a chip-firing game in general. Thus, the total amount of wealth does not change by a legal set-move throughout the game. A vertex with negative dollars is said to be ***debt***. The goal of the game is to have all the vertices of the graph ***out-of-debt*** by a sequence of legal moves. A dollar game is ***winnable*** if it can reach such a state.

In Biggs' invariant [6], a vertex q designated as a government is allowed to go into debt. A government q issues more dollars if and only if no more legal firing is available at vertices on the graph. The states where only q can fire are called to be stable.

Baker's (2015) study [4] proved the following concerning the dollar game on a graph G:

> If the game is winnable, then it can always be won using only borrowing moves. A dollar game is always winnable if the total number of dollars in the game is at least the Euler number. If two configurations are equivalent (meaning that one can get from one to the other by a sequence of legal moves), then they obviously have the same degree (the total number of dollars in the configuration). The converse is false. For each positive integer k, there exist distinct equivalent debt-free configurations

D and E of degree k if and only if G can be disconnected by removing at most k edges. The dollar game is closely related to the famous *Riemann-Roch theorem* in algebraic geometry. (p.2-3)

It turns out that a sequence of chip-firing move does not affect the outcome in the dollar game. Corry & Perkinson [11] refer to this as the abelian property. A commutative diagram in Figure 1.6 illustrates this property.

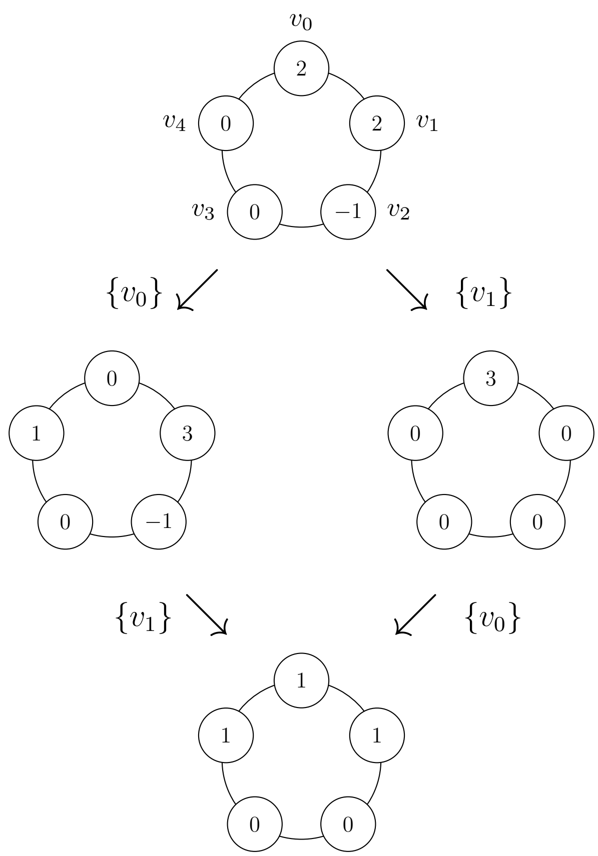

FIGURE 1.6: A commutative diagram of the chip-firing move

Chip-firing move on a graph G can occur not only at a single vertex but also on a set of vertices simultaneously. It is obvious that the reverse chip-firing move from a vertex set $W \subset V$ is the same as a set-chip-firing move from $V - W$.

A divisor on a graph G is

$$\mathrm{Div}(G) = \mathbb{Z}V = \{\sum_{v \in V} f(v)v | f(v) \in \mathbb{Z}\}$$

where $f(v)$ represents the number of chips on a vertex v. In other words, a divisor f represents the distribution of chips on a graph G. The degree of a divisor f equals $\sum_{v \in V} f(v)$.

Two divisors f and f' are 'linearly equivalent' if f' is obtained from f by a sequence of a legal set-move. The divisor class $[f]$ is the collection of linearly equivalent divisors determined by f on G as following

$$[f] = \{f' \in \mathrm{Div}(G) \mid f' \sim f\}.$$

A chip-firing game on a circular graph

According to Corry & Perkinson [11], two divisors f and f' on a circular graph C_n with the same degree are linear equivalent if and only if

$$f \cdot (0, 1, \ldots, n-1) \equiv f' \cdot (0, 1, \ldots, n-1) \mod n.$$

where each f and f' is identified with a vector in $\mathbb{Z}^n$.

We provide the following as a sketch of this bijection.

Proof. ($\Rightarrow$) Identify a divisor f with a vector in $\mathbb{Z}^n$ such that

$$f = (f(v_0), \ldots, f(v_{i-1}), f(v_i), f(v_{i+1}), \ldots, f(v_{n-1}))$$

where $f(v_i) \in \mathbb{Z}$ represents the amount of wealth at a vertex v_i. Then, a chip-firing move at v_i causes a change in the distribution of wealth around v_i. Let f' be a divisor obtained from f by a chip-firing move from v_i such that

$$f' = (f(v_0), \ldots, f(v_{i-1}) + 1, f(v_i) - 2, f(v_{i+1}) + 1, \ldots, f(v_{n-1})).$$

This shows that

$$f \cdot (0, 1, \ldots, n-1) - f' \cdot (0, 1, \ldots, n-1)$$

$$\equiv (i-1) - 2i + (i+1) \equiv 0 \mod n.$$

Therefore,

$$f \cdot (0, 1, \ldots, n-1) \equiv f' \cdot (0, 1, \ldots, n-1) \mod n.$$

($\Leftarrow$) Let a divisor on a circular graph C_n with the degree k be identified with an (n, k)-code in a set $\mathcal{F}_{n,k,t}$ mod n. Provided that two divisors f and f' on a graph C_n satisfy

$$f \cdot (0, 1, \ldots, n-1) \equiv f' \cdot (0, 1, \ldots, n-1) \mod n,$$

the weighted sums of f and f' are equal. This implies that f and f' are in the same set $\mathcal{F}_{n,k,t}$ mod n. Without losing the generality, let f and f' be in $\mathcal{F}_{n,k,0}$. According to the definition of a set $\mathcal{F}_{n,k,0}$ mod n, it is equivalent to a set of the out-of-debt chip-firing game states starting with k chips at a vertex on a circular graph C_n. This means that a code in a set $\mathcal{F}_{n,k,0}$ mod n is obtained from the other in the same set by a legal set-firing move. Therefore, we can conclude that f and f' is linearly equivalent. $\square$

Figure 1.7 illustrates this property. Let f be a divisor representing the leftmost configuration of Figure 1.7. In addition, let f_1 be a divisor obtained from f by firing at v_0 and let f_2 obtained from f_1 by firing on the set of $\{v_0, v_1\}$. Then, each dot products gives that

$$f \cdot (0, 1, \ldots, 4) \equiv (3, 0, 0, 0, 0) \cdot (0, 1, \ldots, 4) \equiv 0 \pmod 5$$

$$f_1 \cdot (0, 1, \ldots, 4) \equiv (1, 1, 0, 0, 1) \cdot (0, 1, \ldots, 4) \equiv 0 \pmod 5,$$

$$f_2 \cdot (0, 1, \ldots, 4) \equiv (0, 0, 1, 0, 2) \cdot (0, 1, \ldots, 4) \equiv 0 \pmod 5.$$

On the contrary, let us find a divisor g with degree 3 satisfies the following.

$$f \cdot (0, 1, \ldots, 4) \equiv g \cdot (0, 1, \ldots, 4) \equiv 0 \pmod 5 .$$

where f is the leftmost divisor in Figure 1.7. If we identify g with a vector (a, b, c, d, e) in $\mathbb{Z}^5$,

$$(3, 0, 0, 0, 0) \cdot (0, 1, \ldots, 4) \equiv (a, b, c, d, e) \cdot (0, 1, \ldots, 4) \equiv 0 \pmod 5 .$$

Then, $b+2c+3d+4e \equiv 0 \pmod 5$ and $a+b+c+d+e=3$. We see that $a=1,\ b=1,\ c=0,\ d=0,\ e=1$ is a solution satisfying both equations. Then, g is identical to f_1 of Figure 1.7. This immediately means that $g \in [f]$ as $f_1 \in [f]$.

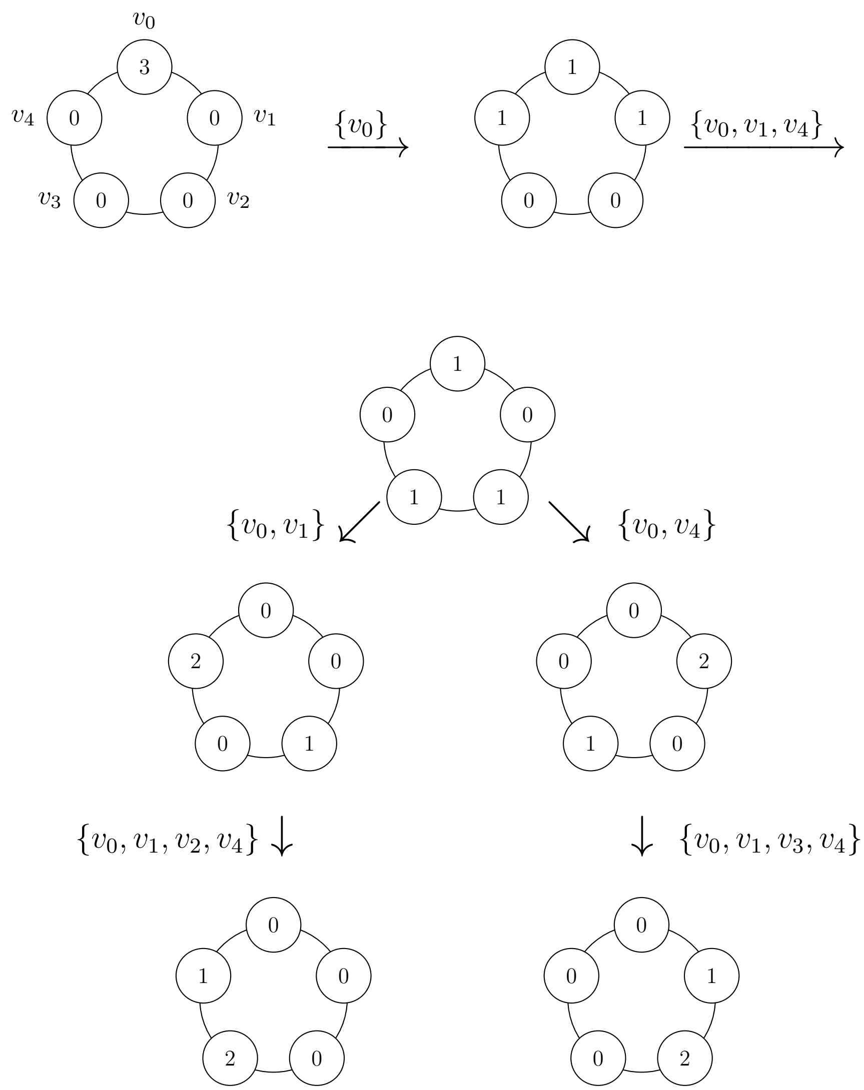

FIGURE 1.7: Several equivalent divisors on C_5

A bijection for the out-of-debt chip-firing states

We already mentioned that a set $\mathcal{F}_{n,k,t}$ of (n,k)-codes mod n is equivalent to a set of out-of-debt chip-firing game states starting with k chips at a vertex on a circular graph C_n. Now let us show how it is obtained.

Recall that the total number of chips and the weighted sum given by a divisor on a circular graph C_n are preserved throughout a sequence of set-firings on vertices. Besides, the codes whose weighted sums are equal belong to the same set $\mathcal{F}_{n,k,t}$ mod n (Corry & Perkinson, [11]). At first, take into account a divisor f in an out-of-debt chip-firing game on C_n starting with k chips on a fixed vertex. Additionally, let $f(i)$ stand for the number of chips piled at a vertex v_i, then a vector in $\mathbb{Z}^n$ identified with f consists of non-negative integers. On that ground we can denote f such that

$$\sum_{i\in[n]} f(i) = k, \quad f = (k, 0, \ldots, 0), \text{ and } f : [n] \to \mathbb{Z}^{\geq 0}.$$

Since each legal move on f does not give negative value at any vertex, every linearly equivalent divisor f' obtained from f by chip-firings also has non-negative values on each vertex. It is certain that the number of chips on the configuration of f' is also k. Consequently,

$$f' : [n] \to \mathbb{Z}^{\geq 0} \text{ and } \sum_{i\in[n]} f'(i) = k \text{ for } f' \in [f].$$

The representation $f = (k, 0, \ldots, 0)$ implies that the weighted sum given by f on C_n is 0. According to a bijection of Corry & Perkinson [11], each element of the divisor class $[f]$ has the same weighted sum 0. Thus, we can go further for a set of divisors equivalent to f as following:

$$\left\{f' \middle| \sum_{i\in[n]} f'(i) = k, \ \sum_{i\in[n]} i \cdot f'(i) \equiv 0 \pmod{n}\right\} \text{ where } f' : [n] \to \mathbb{Z}^{\geq 0}.$$

This representation is the same as the definition of a set $\mathcal{F}_{n,k,0}$ of (n, k)-codes mod n. This explicitly shows that there is a bijection between $\mathcal{F}_{n,k,0}$ and the out-of-debt chip-firing states starting with k chips on a fixed vertex on C_n.

Let f be a divisor corresponding to the leftmost configuration of Figure 1.7. Then, Figure 1.7 shows the divisor class $[f]$ obtained by a set-firing move whereas Figure 1.8 illustrates $\mathcal{F}_{5,4,0}$. It is observed that these two sets are equivalent.

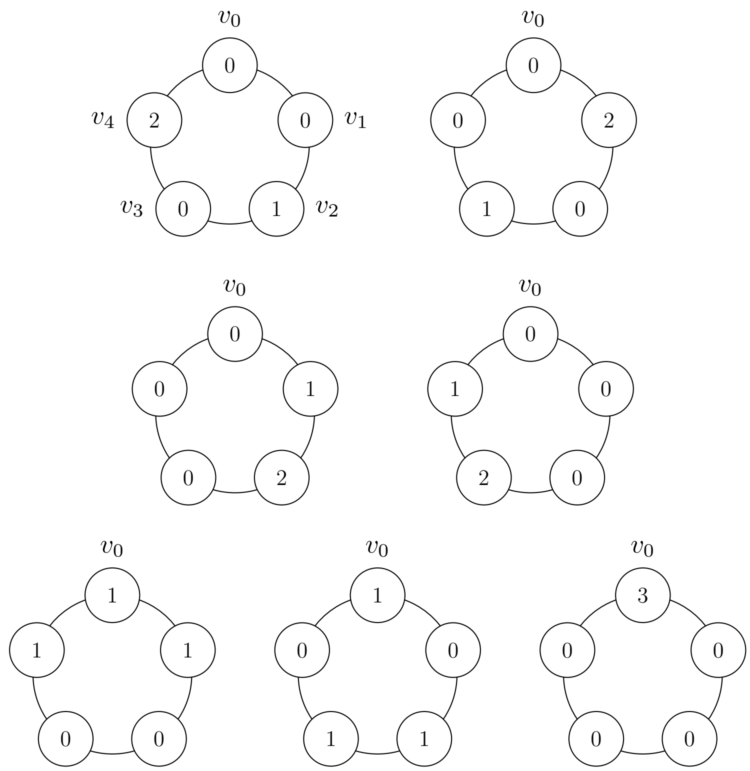

FIGURE 1.8: Every code in $\mathcal{F}_{5,3,0}$

II. CODES AND SLIMES

Slimes

We always envision $[n] := \{1, \ldots, n\}$ to be having the cyclic structure of $\mathbb{Z}_n$. A cyclic interval $[i, j]$ in $[n]$ denotes $\{i, i+1, \ldots, j\}$ in $\mathbb{Z}_n$. All intervals we consider in this thesis will be cyclic intervals. Let f be an (n, k)-code and let m_f denote the largest among sum of two (position-wise) consecutive entries in f. For a cyclic interval $[i, j]$ in $[n]$ of size at least 2, the corresponding sequence $f_i, \ldots, f_j$ is a ***weak-slime*** of f if $f_i + f_{i+1} = \cdots = f_{j-1} + f_j = m_f$. A weak-slime is a ***slime*** if $f_{i-1} + f_i$ and $f_j + f_{j+1}$ are both strictly less than m_f (that is, if it is inclusion-wise maximal among weak-slimes). Notice that slime has size at least 2 according to its definition. Any weak-slime has to have its entries alternating: it has to be of form $a, b, a, b, \ldots, a$ or $a, b, a, b, \ldots, b$ unless they are constant codes.

For instance, for a $(11, 11)$-code f representing the configuration of Figure 2.1, if we index the topmost position with 1 and keep on doing clockwise, $\{11, 1\}, \{2, 3\}, \{3, 4\}, \{2, 3, 4\}, \{7, 8\}, \{8, 9\},$ and $\{7, 8, 9\}$ are weak slimes of f. Among these weak slimes, only $\{11, 1\}, \{2, 3, 4\},$ and $\{7, 8, 9\}$ are slimes since

$$f_2 + f_3 = f_3 + f_4 = 3 \text{ and } f_7 + f_8 = f_8 + f_9 = 3 \text{ where } m_f = 3.$$

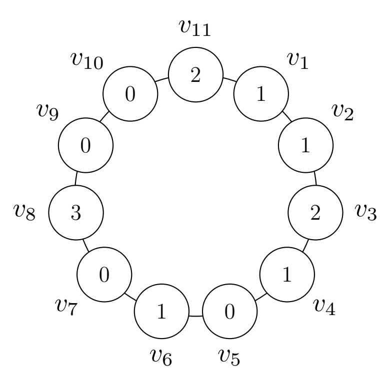

Figure 2.1: Slimes of a code

Given a slime s of size l, its ***weight*** $w(s)$ is defined as $\lfloor \frac{l}{2} \rfloor$. We denote the weight of the code $w(f)$ to be the sum of the weights of all slimes inside the code. Thus, for f in Figure 2.1 $w(f) = \lfloor \frac{2}{2} \rfloor + \lfloor \frac{3}{2} \rfloor + \lfloor \frac{3}{2} \rfloor = 3$.

We say that the slime is ***invalid*** if it is the entire $[n]$ without a cutoff: the sequence $f_1, \dots, f_n$ is an invalid slime when $f_i + f_{i+1} = m_f$ for all $i \in [n]$. We say that the code is ***valid*** if it doesn't contain an invalid slime. A constant code (a code where $f_1 = f_2 = \cdots = f_n$) would have an invalid slime $[n]$ (without a cutoff) and hence be an invalid code. Since odd slimes have to be of form $a, b, \dots, a$ with same entries on its endpoints, it is not hard to see the following:

Lemma 2.1. *When n is odd, the only invalid codes of length n are the constant codes.*

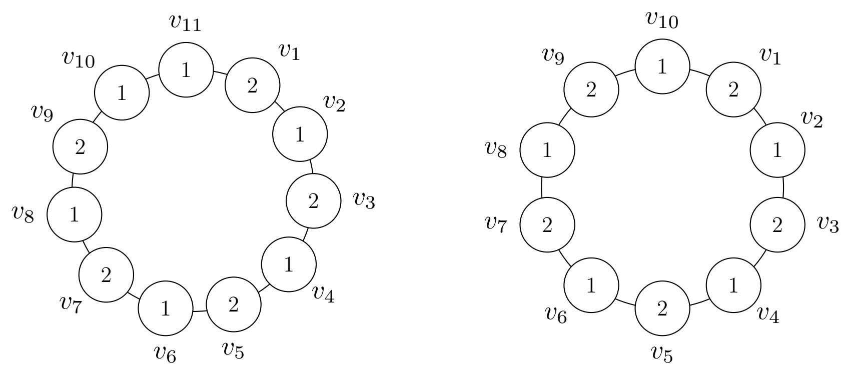

FIGURE 2.2: A code of length 11 vs. A code of length 10

The length of the left code in Figure 2.2 is 11 and it has $\{10, 11\}$ as its slime. Since there is a cutoff between v_{10} and v_{11}, the left code is valid. The weight of it is 5 which is the maximum weight of codes whose lengths are 11. Hence, the only codes which are invalid when the lengths of codes are odd are constant. Whereas the right code whose length is 10 is not constant but there is no cutoff in it. Thus, the right code is invalid. This infers that a code whose length is even might be invalid even when it is not a constant code.

Remark 2.1. Given an (n, k)-code f, its weight $w(f)$ and weighted sum $\sum i \cdot f(i) \pmod n$ are different. The weighted sum changes by k upon rotation. The weight, on the other hand, does not change upon rotation. In other words, the weighted sum is not rotation invariant whereas the weight is rotation invariant.

For instance, the weights of codes of Figure 2.3 are all $\lfloor \frac{2}{2} \rfloor + \lfloor \frac{2}{2} \rfloor = 1$.

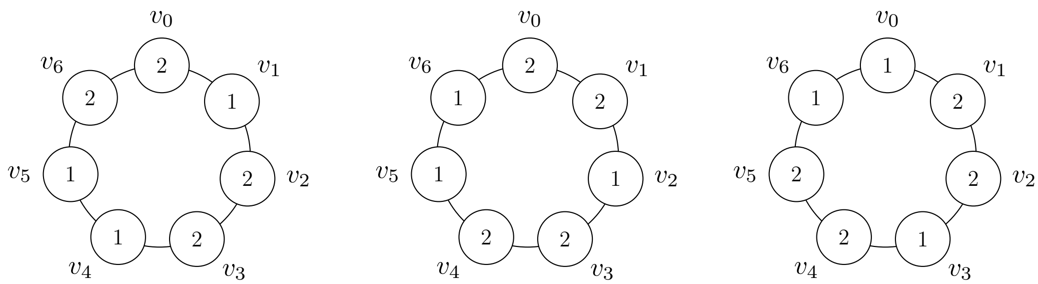

FIGURE 2.3: (7, 11)-codes

However, the weighted sum of the leftmost code is 4, of the middle code is 1, and of the rightmost code is 5 modulo 7.

Slime Migrations

Given a valid slime that has even size, it has to be of the form

$$a, b, \ldots, a, b.$$

The ***(forward) move*** on this slime transforms it to

$$a-1, b+1, \ldots, a-1, b+1,$$

whereas the ***backward move*** transforms the sequence to

$$a+1, b-1, \ldots, a+1, b-1.$$

These moves are well-defined since neither a nor b can be 0. Otherwise, a or b will be m_f and the sequence can't have both a and b at its endpoint to be a slime. The move transforms a slime into a weak slime which can be extended to slime by potentially adding one more position to the right (left for a backward move).

For a valid slime that has an odd size, it has to be of the form

$$a, b, \ldots, a, b, a.$$

The ***(forward) move*** on this slime transforms the sequence to

$$a, b-1, a+1, b-1, \ldots, a+1, b-1, a+1,$$

whereas the ***backward move*** transforms the sequence to

$$a+1, b-1, \ldots, a+1, b-1, a.$$

The move cuts off the leftmost element (rightmost element for a backward move) and the resulting weak slime can be extended to a slime by potentially adding one more position to the right (left for a backward move).

Given an (n, k)-code f, let $\phi_{\rightarrow}(f)$ be the code you get from f by doing a forward move on all slimes of f at the same time. We call this the ***(forward) migration*** of all slimes. Similarly, let $\phi_{\leftarrow}(f)$ be the code you get from f by doing a backward move on all slimes of f at the same time and call this the ***backward migration*** of all slimes. The migration changes $\sum i \cdot f(i)$ modulo n by the weight of the code.

Take a look at Figure 2.4. If we do the forward migration on the leftmost code, we get the code in the middle. If we do the forward migration of the code in the middle, we get the rightmost code. If we do the backward migration on the code in the middle, we get the leftmost code back.

Lemma 2.2. *For any valid (n, k)-code f, we have $\phi_{\leftarrow}(\phi_{\rightarrow}(f)) = f$.*

Proof. Any even sized slime s of the form $a, b, \ldots, a, b$ after a forward move becomes either

$$a-1, b+1, \ldots, a-1, b+1 \quad \text{or} \quad a-1, b+1, \ldots, a-1, b+1, a-1$$

The latter case absorbs a new element to the right. In the first case it is obvious the backward move returns it back to s. In the second case since $a + b > a - 1 + b$, the rightmost element gets cut off upon backward migration, so that we get s back as well. A similar analysis holds for odd-sized slimes. $\square$

In Figure 2.4, notice that the weights of all three codes are the same. It is true in general that the migration operation preserves the number of slimes and the total weight as well:

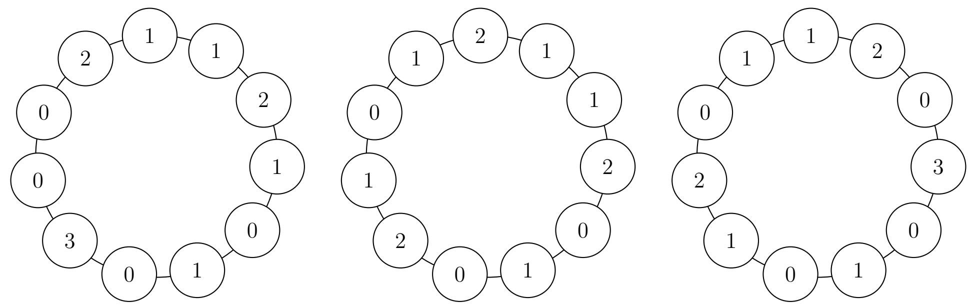

FIGURE 2.4: Slime migrations

Lemma 2.3. *For any valid (n, k)-code f, migration does not change the weight of f. That is, we have $w(\phi_{\leftarrow}(f)) = w(f) = w(\phi_{\rightarrow}(f))$.*

Proof. Given an odd-sized slime, its size is either maintained or decreased by 1 after a movement. Given an even-sized slime, its size is either maintained or increased by 1 after a movement. Hence the weight of each slime is preserved after the migration.

Consider the case when we have two adjacent slimes in f: $f_i, \ldots, f_k$ is a slime and $f_{k+1}, f_{k+2}, \ldots, f_j$ is another slime. After forward migration, if the latter slime was even length then $\ldots, f_k$ and $f_{k+1}, \ldots$ are still separate slimes since $f_k + f_{k+1}$ stays the same and is strictly smaller than m_f. If the latter slime had odd length then f_{k+1} gets cut off from the slime to the right anyways. So there is no fear of two slimes merging after a migration.

Since the weight of each slime is preserved and slimes do not merge nor split, it is explicitly realized that the weight of a code is preserved upon slime migration. □

Given a valid (n, k)-code f that has weight $w(f)$ coprime to n, let $i(f)$ denote the inverse of $w(f)$ modulo n. Define $\phi(f)$ to be the map that sends f to $(\phi_{\rightarrow})^{i(f)}(f)$. Combining what we have so far we get:

Proposition 1. *Suppose that $w(f)$ is coprime to n. Then the map ϕ is invertible and weight preserving. Furthermore, the image of a valid code in $\mathcal{F}_{n,k,t}$ under ϕ is a valid code of $\mathcal{F}_{n,k,t+1}$.*

Proof. Assume that $w(f)$ is coprime to n. Then, $w(f)$ generates $\mathbb{Z}_n$, so that $i(f)$ exists such that $i(f)w(f) = 1$. Let $\phi(f) = (\phi_\rightarrow)^{i(f)}(f)$. As we have already shown, the weight is preserved upon slime migration. Hence, ϕ also preserves the weight. $\phi_\leftarrow(\phi_\rightarrow(f)) = f$, so that

$$(\phi_\leftarrow)^{i(f)}(\phi(f)) = (\phi_\leftarrow)^{i(f)}\big((\phi_\rightarrow)^{i(f)}(f)\big) = f.$$

Therefore $\phi^{-1} = (\phi_\leftarrow)^{i(f)}$ verifying that ϕ is invertible. Upon slime migration the weighted sum increases by $w(f)$. Thus, $i(f)w(f)$ times of migration results increment of 1 modulo n. Thus, ϕ maps f in $\mathcal{F}_{n,k,t}$ to a code in $\mathcal{F}_{n,k,t+1}$. Since the weight is preserved under ϕ and the existence of the weight of a code infers that it is valid, $\phi(f)$ is valid if f is valid. □

Take a look at Figure 2.5 illustrating the Proposition 2.5. Let f be a code referring to the left topmost configuration in Figure 2.5. Then, $f \in \mathcal{F}_{5,6,1}$ and $w(f) = 2$. Since $w(f)$ is coprime to 5, $i(f) = 3$ is obtained. Such conditions yield that $\phi : f \to (\phi_\rightarrow)^3(f)$ defined by in detail

$$f \mapsto (\phi_\rightarrow)^1(f) \mapsto (\phi_\rightarrow)^2(f) \mapsto (\phi_\rightarrow)^3(f) = \phi(f).$$

It is realized that $\phi(f)$ refers to the right topmost configuration of Figure 2.5. One can see that the map ϕ sends a code $f \in \mathcal{F}_{5,6,1}$ to code $\phi(f) \in \mathcal{F}_{5,6,2}$ through three times of forward migration. The natural implication goes in the other direction. Suppose g is a code referring to the right topmost configuration of Figure 2.5. Then, three times of backward migrations on $g \in \mathcal{F}_{5,6,2}$ result to f where $f \in \mathcal{F}_{5,6,1}$. On the other hand, Figure 2.5 illustrates that the slimes does not merge nor split throughout the slime migrations. Thereby, we can see that all codes in Figure 2.5 are valid and the weights of them are preserved:

$$w(f) = w(\phi_{\rightarrow}(f)) = w((\phi_{\rightarrow})^2(f)) = w((\phi_{\rightarrow})^3(f)) = 2,$$

$$w(g) = w(\phi_{\leftarrow}(g)) = w((\phi_{\leftarrow})^2(g)) = w((\phi_{\leftarrow})^3(g)) = 2.$$

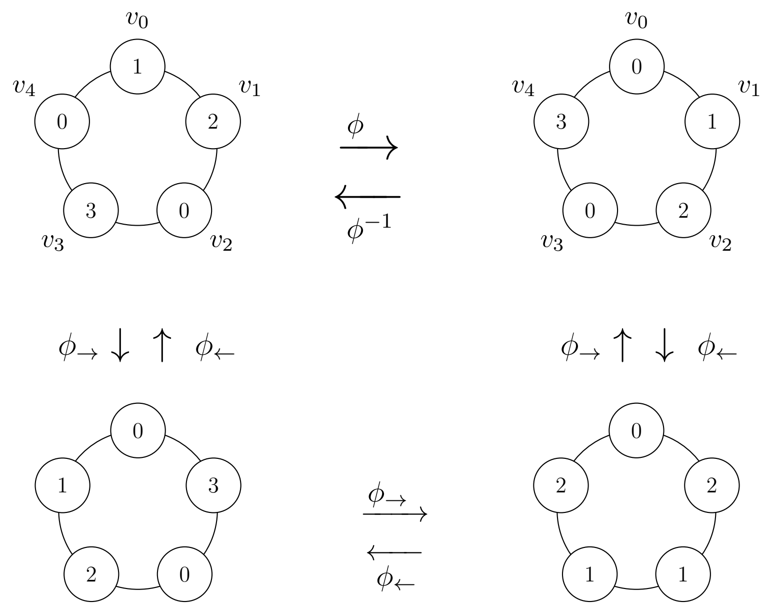

Figure 2.5: A riwi-map via slime migrations

III. BIJECTION AND NECKLACES

We proposed that a certain bijective map between $\mathcal{F}_{n,k,t}$'t induces a bijection $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$ in the general case. To accomplish it, at first label the black beads of a necklace of $\mathcal{N}_{n,k}$ 1 to n in some clockwise order. After that, express a necklace of $\mathcal{N}_{n,k}$ as a sequence $(g_0, \ldots, g_{n-1})$ where g_i represents the number of white beads between black beads labeled i and $i+1$. Since a necklace $g = (g_0, \ldots, g_{n-1})$ is a cyclic rotation invariant class, it is a collection of codes such that

$$g, c(g), \ldots, c^{n-1}(g).$$

We dedicate q such that

$$q = \frac{n}{\gcd(n,k)}.$$

Besides, we call the collection

$$\{f, c^q(f), c^{2q}(f), \ldots, c^{n-q}(f)\}$$

a ***neck-class*** of $\mathcal{F}_{n,k,t}$ for a given code $f \in \mathcal{F}_{n,k,t}$. Explicit conjecture is that every code in a neck-class including $f \in \mathcal{F}_{n,k,t}$ belongs to the same set $\mathcal{F}_{n,k,t}$. This gives the natural implication such that all codes in a neck-class have the same weighted sum modulo n.

Let f and g be the first codes of Figure 3.1 and Figure 3.2, respectively. Then, f is in $\mathcal{F}_{9,6,1}$ and g is in $\mathcal{F}_{9,6,4}$ and $q = 3$ for both f and g. Figure 3.1 and Figure 3.2 shows that $\{f, c^3(f), c^6(f)\}$ and $\{g, c^3(g), c^6(g)\}$ are neck-classes of f and g, respectively, where their weighted sums are preserved (mod 9) upon rotations of period 3.

For a code f of Figure 3.3, $q = \frac{5}{\gcd(5,6)} = 5$. This means that the weighted sum of a code of Figure 3.3 is preserved by rotations of period 5. Therefore, a neck-class of $\mathcal{F}_{5,6,5}$ is $\{f\}$.

Let $\mathcal{F}'_{n,k,t}$ denote the codes of $\mathcal{F}_{n,k,t}$ that have period n. Let $\mathcal{N}'_{n,k}$ denote the necklaces of $\mathcal{N}_{n,k}$ that have period n.

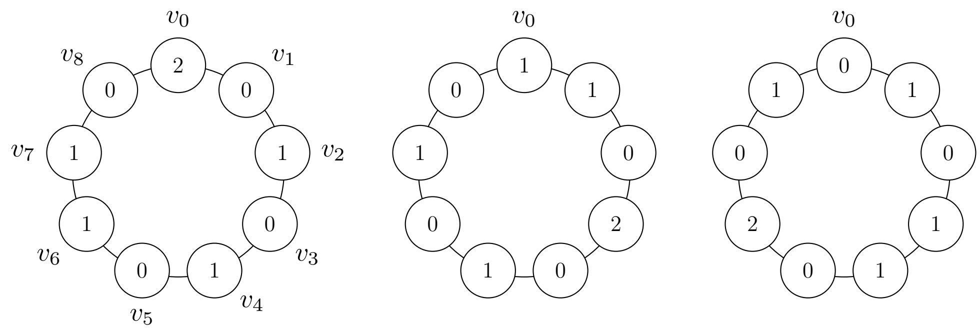

FIGURE 3.1: A neck-class of $\mathcal{F}_{9,6,1}$ of period 3

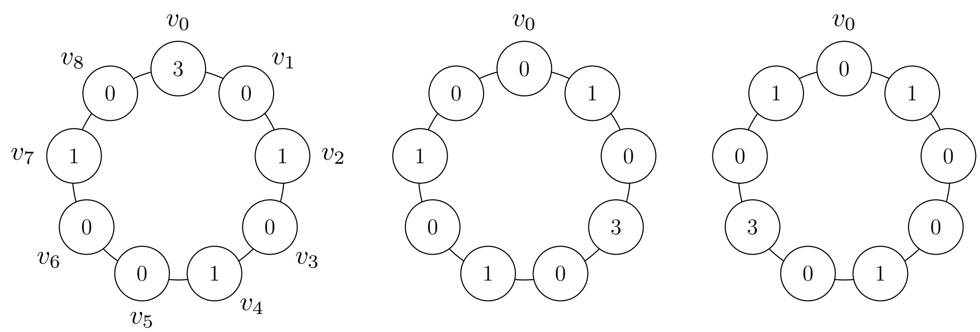

FIGURE 3.2: A neck-class of $\mathcal{F}_{9,6,4}$ of period 3

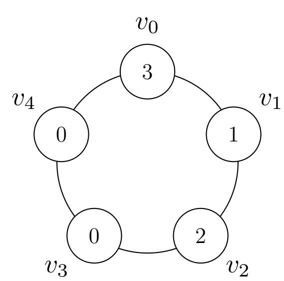

FIGURE 3.3: A neck-class of $\mathcal{F}_{5,6,5}$ of period 5

A Riwi-Map

Definition 3.1. We say that a map $\chi : \mathcal{F}'_{n,k} \to \mathcal{F}'_{n,k}$ is a ***riwi-map*** where $\mathcal{F}'_{n,k} := \bigcup_t \mathcal{F}'_{n,k,t}$ if:

- it is a bijective map,
- (rotation invariant) $c\chi = \chi c$, and
- (weighted sum increasing) if $f \in \mathcal{F}'_{n,k,t}$ then $\chi(f) \in \mathcal{F}'_{n,k,t+1}$.

Using a riwi-map χ we can construct a map σ_χ between $\mathcal{N}'_{n,k}$ and $\mathcal{F}'_{n,k,0}$ in the following way: for each neck-class in $\mathcal{F}_{n,k,0}$, fix an arbitrary representative f. Let σ_χ be a map from $\mathcal{F}'_{n,k,0}$ to $\mathcal{N}'_{n,k}$ that sends $c^{iq}(f)$ to $\chi^i(f)$ for each $0 \le i < \frac{n}{q}$.

Theorem 3.1. *When χ is a riwi-map, the map σ_χ is a bijection between $\mathcal{F}'_{n,k,0}$ and $\mathcal{N}'_{n,k}$.*

Proof. We first show that the map $\sigma_\chi : \mathcal{F}'_{n,k,0} \to \mathcal{N}'_{n,k}$ is one-to-one. Assume for sake of contradiction that the image of some $c^{iq}(f)$ and $c^{jq}(g)$ are the same. Due to $\chi^i(f) \in \mathcal{F}'_{n,k,i}$, we must have $i = j$. But since χ is a bijective map, $\chi^i(f) = \chi^i(g)$ implies $f = g$ and we get a contradiction.

Next, we show that the map is onto. Pick any necklace in $\mathcal{N}'_{n,k}$. Choosing a position here gives a code g in $\mathcal{F}'_{n,k,j\frac{n}{q}+i}$ for some $0 \le j < q$ and $0 \le i < \frac{n}{q}$. We can replace g with a rotation equivalent code in $\mathcal{F}'_{n,k,i}$. Take the neck-class in $\mathcal{F}'_{n,k,i}$ containing g. Thanks to χ being rotation invariant, applying $(\chi^{-1})^i$ on the neck-class gives a neck-class in $\mathcal{F}'_{n,k,0}$. Pick f to be the chosen representative of that neck-class. Then, $c^{iq}(f)$ is mapped to $\chi^i(f)$ under σ_χ which is rotation equivalent to g. $\square$

Consider a necklace of $\mathcal{N}'_{6,10}$. Then, it has the period 3 since $q = \frac{6}{\gcd(6,10)} = 3$. This tells that a given code g to a necklace of $\mathcal{N}'_{6,10}$ has a pair g' obtained by rotating g three times where their weighted sums are equivalent modulo 6.

Take one from each pair and let them be $g_1,\ g_2$, and g_3. Then, the configurations in the first row of Figure 3.4 correspond to $g_1,\ g_2$, and g_3. Let $g'_1,\ g'_2$ and g'_3 be the rest of each pair, then the configurations in the second row of Figure 3.4 correspond to them, respectively. The fact that $\frac{n}{q} = 2$ facilitates a coding by that each of $g_1,\ g_2$, and g_3 belongs to a $\mathcal{F}'_{6,10,2j+i}$ for some $0 \le j < 3$ and $0 \le i < 2$. On the other hand, Figure 3.5 shows that $\{g_1,\ g'_1\}$, $\{g_2,\ g'_2\}$, and $\{g_3,\ g'_3\}$ are three neck-classes obtained from a necklace of $\mathcal{N}_{6,10}$ in Figure 3.4. Moreover, we can see that $g'_1 \in \mathcal{F}'_{6,10,2}$, $g'_2 \in \mathcal{F}'_{6,10,0}$, and $g'_3 \in \mathcal{F}'_{6,10,4}$.

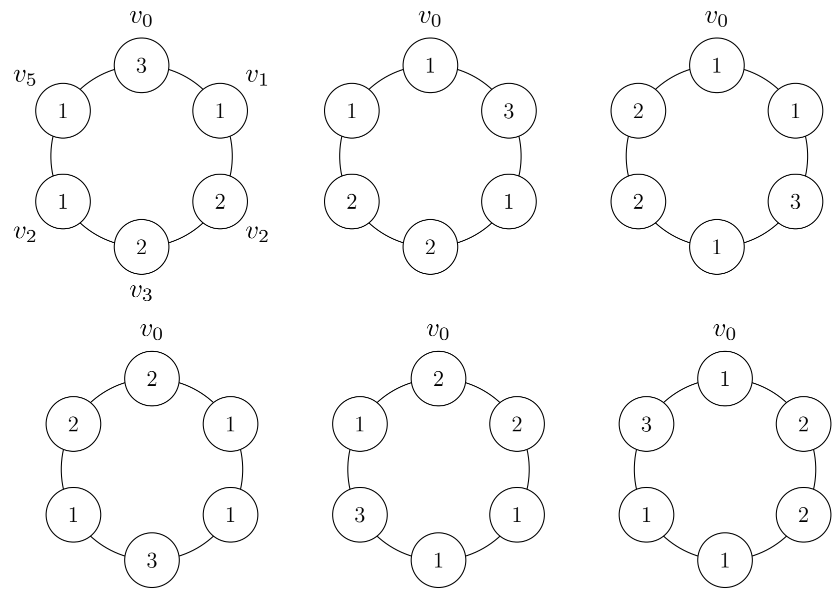

FIGURE 3.4: A necklace of $\mathcal{N}'_{6,10}$

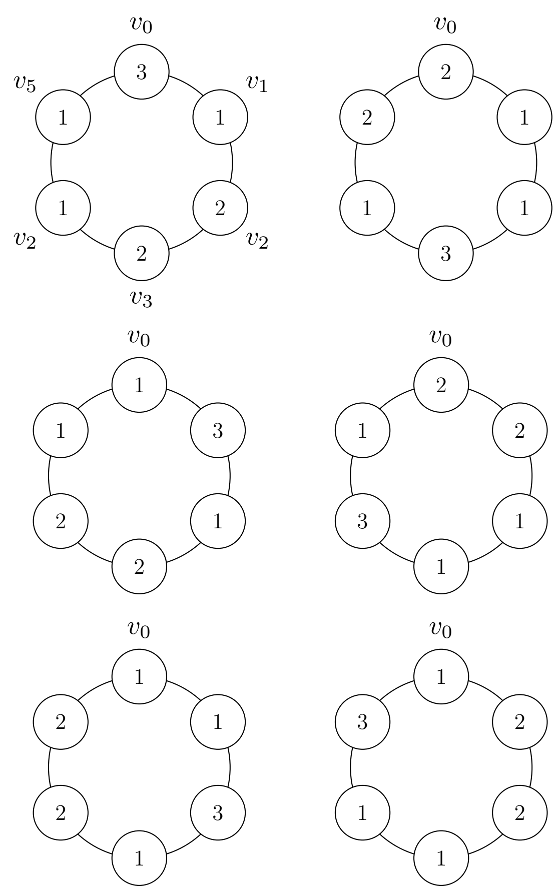

FIGURE 3.5: Neck-classes obtained from a necklace of $\mathcal{N}'_{6,10}$

Now, assume that there exists a riwi-map χ sending a code of $\mathcal{F}'_{6,10,t}$ to a code of $\mathcal{F}'_{6,10,t+1}$. Thereby, a code $f \in \mathcal{F}'_{6,10,0}$ is surely found by applying the inverse of χ several times on the codes $g'_1,\ g'_2$ or g'_3 as following

$$(\chi^{-1})^2(g'_1) = f \in \mathcal{F}'_{6,10,0},$$

$$(\chi^{-1})^0(g'_2) = f \in \mathcal{F}'_{6,10,0},$$

$$(\chi^{-1})^4(g'_3) = f \in \mathcal{F}'_{6,10,0}.$$

Next, consider a neck-class $\{c^{3\cdot 2}(f),\ c^{3\cdot 0}(f),\ c^{3\cdot 4}(f)\}$ that is obtained by rotating f $3i$ times where i is taken from $g' \in \mathcal{F}_{6,10,i}$.

Finally, a bijective map

$$\sigma_\chi : \mathcal{F}_{6,10,0} \mapsto \mathcal{N}_{6,10} \ \text{ defined by } \ c^{3i}(f) \mapsto (\chi^{-1})^i(g').$$

is constructed extremely depending on a riwi-map χ.

Let us unfold a path from a code g representing any configuration of a necklace of $\mathcal{N}_{6,10}$ in Figure 3.4 to a code in $\mathcal{F}_{6,10,0}$ through a neck-class $\{g, g'\}$ in Figure 3.5:

$$g \mapsto g' \mapsto (\chi^{-1})^i(g') \mapsto f \mapsto c^{3i}(f).$$

This path clearly shows that the map $\sigma_\chi : \mathcal{F}_{6,10,0} \mapsto \mathcal{N}_{6,10}$ is surjective under the assumption that there exists a riwi-map such that $\chi^i(f) = g' \in \mathcal{F}'_{6,10,i}$. With the same assumption, look at the following two different paths starting from the codes in $\mathcal{F}_{6,10,0}$ ending at the codes given to a necklace of $\mathcal{N}_{6,10}$:

$$c^{3\cdot 2}(f) \mapsto f \mapsto (\chi^{-1})^2(g'_1) \mapsto g'_1 \mapsto g_1$$

$$c^{3\cdot j}(f') \mapsto f' \mapsto (\chi^{-1})^j(h') \mapsto h' \mapsto h$$

Consider g_1 is the code representing the left topmost configuration of Figure 3.5. As for the injectivity of σ_χ, assume that $(\chi^{-1})^2(g'_1) = (\chi^{-1})^j(h')$ whereas $c^{3\cdot 2}(f) \neq c^{3\cdot j}(f')$. Then, since χ is a riwi-map, g_1 and h should be in the same neck-class as seen in the first row of Figure 3.5. However, there is no way for h to be in the same neck-class with g_1 unless $2 = j$. This obviously contradicts that $c^{3\cdot 2}(f) \neq c^{3\cdot j}(f')$.

For codes and necklaces of period $p < n$ in $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$ (here p has to be a common divisor of n and k), we can extend the bijection between $\mathcal{N}'_{\frac{n}{p},\frac{k}{p}}$ and $\mathcal{F}'_{\frac{n}{p},\frac{k}{p},0}$. Hence the problem of constructing a bijection between $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$ can be reduced to the problem of finding a riwi-map on $\mathcal{F}'_{n,k}$'s.

Bijection when n and k are Coprime

It was proved in Theorem 5.2 by Brauner et al. [9] that there is a bijection between $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$ when n and k are coprime. They proved this with the perspective of equivalent the divisors on a circular graph C_n. Another way to think of this is that one can simply take a certain power of the rotation map as the riwi-map.

Corollary 3.1. *When n and k are coprime, Theorem 3.1 gives a bijection between $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$.*

Proof. When n and k are coprime, every code and necklace have period n. So we have $\mathcal{N}_{n,k} = \mathcal{N}'_{n,k}$ and $\mathcal{F}_{n,k} = \mathcal{F}'_{n,k}$. Pick $i(k)$ to be the inverse of k modulo n. Then, $c^{i(k)}$ is a riwi-map since rotating $i(k)$ times increases the weighted sum of a code by $k \cdot i(k) \equiv 1 \pmod{n}$. $\square$

Give codes f_1, f_2 and f_3 to a necklace of $\mathcal{N}_{3,5}$ in Figure 3.6 in a row. Then, $f_1 \in \mathcal{F}_{3,5,2}$, $f_2 \in \mathcal{F}_{3,5,1}$, and $f_3 \in \mathcal{F}_{3,5,0}$. For $(3,5)$-codes, $i(5) = 2$ since $5 \cdot 2 \equiv 1 \pmod{3}$. Therefore, it is clear that c^2 sends a code in $\mathcal{F}_{3,5,t}$ to a code $\mathcal{F}_{3,5,t+1}$. A rotation map is bijective when n and k are coprime, so is c^2 as long as n and k are coprime. Besides, it is taken granted that c^2 is rotation invariant. Hence, we can conclude that $c^2 : \mathcal{F}_{3,5} \to \mathcal{F}_{3,5}$ is a riwi-map. Next, pick a code f in $\mathcal{F}_{3,5,0}$ that is identical to f_3 in Figure 3.6. Then, a neck-class including f is a singleton set $\{f\}$ as shown in Figure 3.7 since $q = n = 3$. Now, we can have a bijective map σ_χ sending a code given to a necklace of $\mathcal{N}_{3,5}$ in Figure 3.6 and a code in a neck-class in Figure 3.7 defined by

$$f_1 \in \mathcal{N}_{3,5} \mapsto \chi^1(f_1) = f_3 = f \mapsto c^{1\cdot 3}(f) \in \mathcal{F}_{3,5,0},$$

$$f_2 \in \mathcal{N}_{3,5} \mapsto \chi^2(f_2) = f_3 = f \mapsto c^{2\cdot 3}(f) \in \mathcal{F}_{3,5,0},$$

$$f_3 \in \mathcal{N}_{3,5} \mapsto \chi^0(f_3) = f_3 = f \mapsto c^{0\cdot 3}(f) \in \mathcal{F}_{3,5,0}.$$

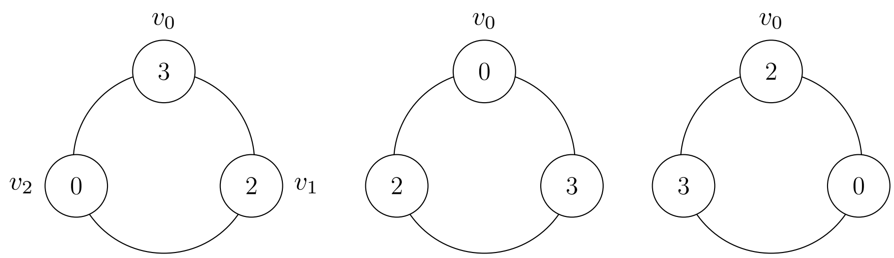

FIGURE 3.6: A necklace of $\mathcal{N}_{3,5}$

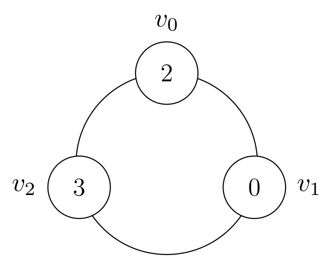

FIGURE 3.7: A neck-class of $\mathcal{F}_{3,5,0}$

As for the necklaces of $\mathcal{N}_{2,k}$ with k is odd, we can also simply use a rotation map c as a riwi-map χ since n and k are coprime. For instance, let f be a code representing a necklace of $\mathcal{N}_{2,k}$ in form of (a, b), then send it under a riwi-map χ

$$\chi : (a, b) \mapsto (b, a).$$

Since $a + b = k$ and k is odd, a and b cannot have the same parity. Hence, if we assume that (a, b) is in $\mathcal{F}_{2,k,t}$, (b, a) must be in $\mathcal{F}_{2,k,t+1}$.

Bijection when n and k are Not Coprime

In the case n is an odd prime, we use the slime migration map ϕ as our riwi-map.

Corollary 3.2. *When n is an odd prime, Theorem 3.1 gives a bijection between $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$.*

Proof. When n is an odd prime, all codes and necklaces have period 1 or n. Lemma 2.1 tells us that the only invalid codes are the constant codes. Associating the constant codes to constant necklaces, all that remains is to show that ϕ of Proposition 1 is a riwi-map. Bijection and weighted sum increasing are already done, so we need to check rotation invariance. The forward migration map $\phi_{\rightarrow}$ does not depend on any choice of a position on the circle since slimes are defined using sums of adjacent entries. Therefore, ϕ is rotation invariant as well. □

Pick the leftmost necklace of $\mathcal{N}_{3,3}$ in Figure 3.8 and give it a code f. Since $w(f) = 1$ and it is coprime to 3, $i(f) = 1$. Thus, we have the slime migration such that $\phi = \phi_{\rightarrow}$. Figure 3.8 shows that the slime migration changes the leftmost code f to the middle code $\phi(f)$. Doing it one more time gives the rightmost code $\phi^2(f)$. Each of them belongs to a different $\mathcal{F}_{3,3,t}$ since the weighted sum changes upon slime migrations. Moreover, they are rotation variant, so that each of them represents a different necklace of $\mathcal{N}_{3,3}$.

$$f \in \mathcal{F}_{3,3,0} \mapsto \phi(f) \in \mathcal{F}_{3,3,1} \mapsto \phi^2(f) \in \mathcal{F}_{3,3,2}.$$

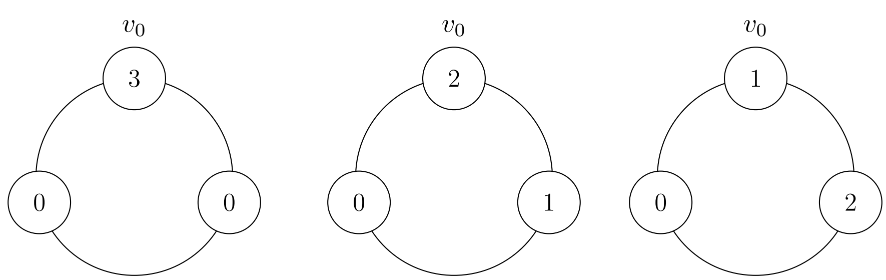

FIGURE 3.8: Slime migrations on a code f given to a necklace $\mathcal{N}_{3,3}$

Rotate f and then make a slime migration on it and vice versa. Both cases yield the same result, that is, the rotated second code of Figure 3.8. Therefore, it is clearly stated that the slime migration on $f \in \mathcal{F}_{3,3,t}$ satisfies three conditions of a riwi-map in the case n and $w(f)$ are coprime.

Figure 3.9 shows a neck-class $\{f, c(f), c^2(f)\}$ of $\mathcal{F}_{3,3,0}$ including a code f. Since (n, k)-codes are not rotation invariant, a code in a neck-class $\{f, c(f), c^2(f)\}$ is different from the other.

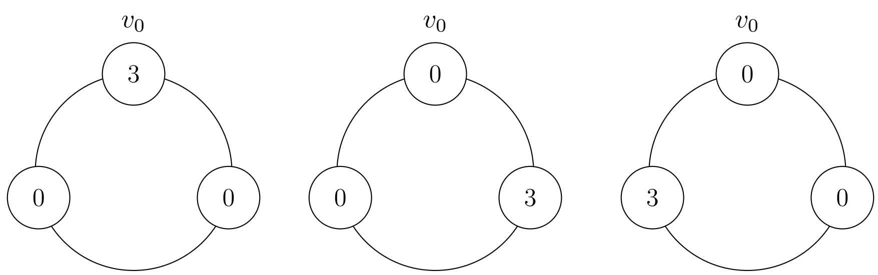

FIGURE 3.9: A neck-class of $\mathcal{F}'_{3,3,0}$ including f

Now, map the codes in Figure 3.9 to the codes in Figure 3.8 under a bijective map σ_ϕ defined by

$$\{f, c(f), c^2(f)\} \mapsto \{f, \phi(f), \phi^2(f)\}.$$

Notable thing is that a bijective map σ_ϕ sends a set of three different codes of the neck-class including a code in $\mathcal{F}_{3,3,0}$ to a set of codes given to three different necklaces in $\mathcal{N}_{3,3}$. This is viable since $w(f)$ is coprime to an odd prime n.

Theorem 3.2. *We can construct a bijection between $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$ when n is prime.*

Proof. We only need to consider the case when $n = 2$ and k is even. For a necklace of form a, b with $a \geq b$, map it to the code a, b if b is even. Otherwise map it to $b - 1, a + 1$. $\square$

In the case k is odd, a rotation map works as a riwi-map so that it gives a bijection between $\mathcal{N}_{2,k}$ and $\mathcal{F}_{2,k,0}$ since k is coprime to 2. However, the slime migration does not when $n = 2$ and k is even. The reason our argument does not work directly for n that is not an odd prime, is that the slime migration is not guaranteed to be a bijection between $\mathcal{F}'_{n,k,t}$'s. For example if we do a slime migration on $2, 0$, we get $1, 1$. However, $1, 1$ is also obtained from $0, 2$ by the same migration. Thus, a slime migration is not bijective when $n = 2$ and it can't be a riwi-map as a result. This is why we have to take care of $n = 2$ case separately.

Instead, a map described in the proof of Theorem 3.2 gives a direct bijection. Define $\sigma : \mathcal{N}_{2,6} \to \mathcal{F}_{2,6,0}$ to be such a map. Then, Figure 3.10 demonstrates that each code given to four different necklaces of $\mathcal{N}_{2,6}$ is sent to one of four different codes in $\mathcal{F}_{2,6,0}$. Besides, one can see that there exists a code given to a necklace of $\mathcal{N}_{2,6}$ to which a code of $\mathcal{F}_{2,6,0}$ is mapped by the map σ^{-1}.

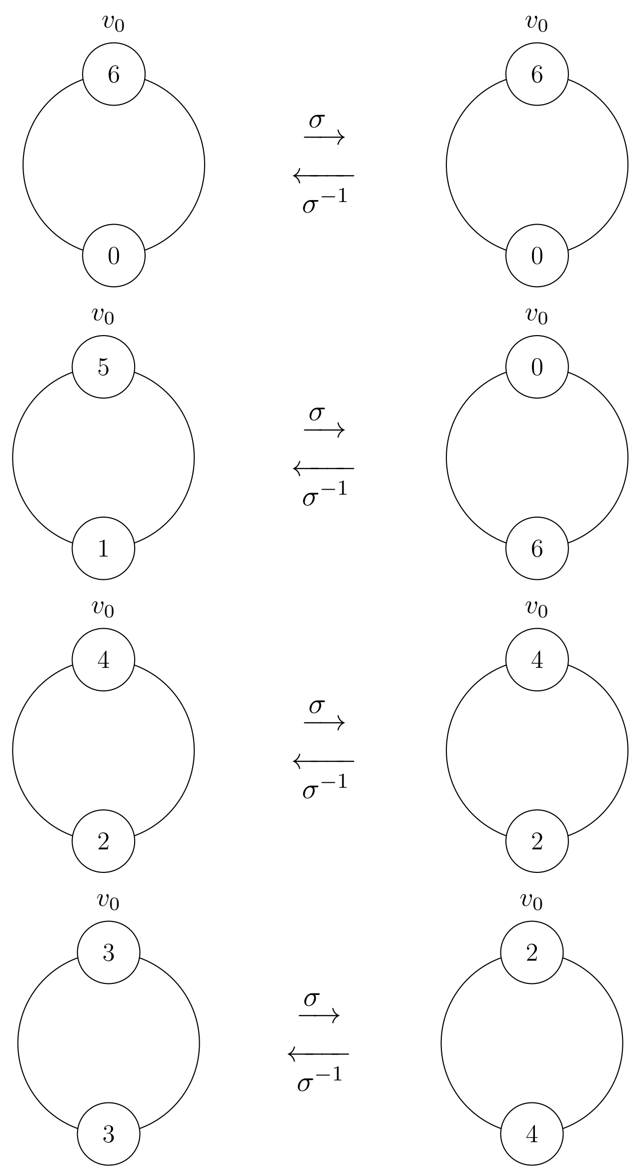

FIGURE 3.10: A bijection between $\mathcal{N}_{2,6}$ and $\mathcal{F}_{2,6,0}$

By showing a bijection when $n = 2$, we have come to the end of constructing a new bijection between $\mathcal{F}_{n,k,0}$ and $\mathcal{N}_{n,k}$ in the case n is prime.

IV. FUTURE DIRECTION FOR RESEARCH

We complete the construction of a new bijection between $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$ when n is prime. A riwi-map provides a fundamental ground for this bijection when n is an odd prime. More specifically speaking, it is a slime migration that works as a riwi-map between $\mathcal{F}_{n,k,t}$'s when k is not coprime to an odd integer n. By building up a direct bijection between $\mathcal{F}_{2,k,0}$ and $\mathcal{N}_{2,k}$ in the case that k is even, all the work to constitute a new bijection between $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$ in the case n is prime comes to the end. However, generalizing is not completed yet.

Question 1. Can one construct a bijection between $\mathcal{N}_{n,k}$ and $\mathcal{F}_{n,k,0}$, for general n?

The strategy would be to find the riwi-maps using Theorem 3.1. A good candidate is a modification of the slime migration map ϕ. Notice that ϕ working as a riwi-map only depends on the weight of the codes being coprime to n, instead of what k is. Since ϕ preserves the weight of the code, we can refine $\mathcal{N}'_{n,k}$ further based on the weight, and separately take care of the cases when the weight isn't coprime to n. Using this idea for small numbers like $n = 4$ and $n = 6$, it is pretty straightforward to construct a riwi-map using the fact that there are not many codes with $w(f) = 2$ (being 3 is impossible) and hence get a bijection easily.

REFERENCES

[1] Anderson, R., Lovász, L., Shor, P., Spencer, J., Tardos., & Winograd, S. (1989). Disk, balls, and walls: analysis of a combinatorial game. *American Mathematics*, 96 (6).

[2] Ardila, F., Castillo, F., & Henley, M. (2014). The Arithmetic Tutte Polynomials of the Classical Root Systems. *International Mathematics Research Notices*, 2015(12):3830-3877.

[3] Bacher, R., Harpe, P., & Nagnibeda, T. (1997). The lattice of integral flows and the lattice of integral cuts on a finite graph. *Bulletin de la Société Mathématique de France*, 125 (2), 167-198.

[4] Baker, M. (2015). The Dollar Game on a Graph (with 9 surprises!). Retrieve from https://people.math.gatech.edu/ mbaker/pdf/g4g9.pdf

[5] Berstel, J. & Perrin, D. (2007). The origins of combinatorics on words. *European Journal of Combinatorics*, 28.

[6] Biggs, N. (1999). Chip-Firing and the Critical Group of a Graph. *Journal of Algebraic Combinatorics*, 9, 25-45.

[7] Björner, A. & Lovász, L. (1992). Chip-firing games on directed graphs. *Journal of Algebraic Combinatorics*, 1, 305-328.

[8] Björner, A., Lovász, L., & Shor, P. (1991). Chip-firing games on graphs. *European Journal of Combinatorics*, 12, 283-291.

[9] Brauner, S., Glebe, F., & Perkinson, D. (2019). Enumerating linear systems on graphs. *ArXiv Mathematics e-prints* .

[10] Chan, S. (2019). A bijection between necklaces and multisets with divisible subset sum. *Electronic Journal of Combinatorics*, 26(1): P1.37.

[11] Corry, S. & Perkinson, D. (2018). *Divisors and Sandpiles: An Introduction to Chip-Firing*. MAA reviews. American Mathematical Society.

[12] Erdös, P. & Heilbronn, H. (1964). On the addition of residue classes mod p, *Acta Arithmetica*, 9, 149-159.

[13] Kitchloo, N. & Pachter, L.(1993). An interesting result about subset sums. MIT Unpublished Notes.

[14] Klivans, C. (2018). *The Mathematics of Chip-firing*. CRC Press.

[15] Spencer, J. (1986). Balancing vectors in the max norm. *Combinatorica*, 6(1).

[16] Stanley, R. (1986). *Enumerative Combinatorics*, Volume 1, (2^{nd} edition). Cambridge Studies in Advanced Mathematics.