HÁSKÓLINN Í REYKJAVÍK
REYKJAVIK UNIVERSITY

ICE-TCS
Icelandic Centre of Excellence
in Theoretical Computer Science

Combinatorial Structures for CSP

Duncan Adamson
Reykjavik University, Department of Computer Science

May 24, 2022

Outline

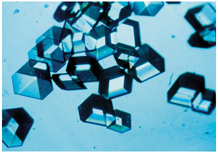
This Talk is split into three sections:

1. Background on both the chemical motivation behind this talk, and 1D necklaces.
2. An introduction to **multidimensional necklaces** as a combinatorial object.
3. A set of results on several fundamental problems associated with 1D necklace objects when generalised to the multidimensional setting.

Why Crystals?

- New materials are needed to deal with the challenges of the 21st century, from strong materials for manufacturing to better conductors for electrical systems.
- Crystals are a fundamental, and very common form of matter.
- Importantly, Crystals are **periodic** - meaning that a lot of the properties of a crystaline material can be determined from a relatively small amount of information.

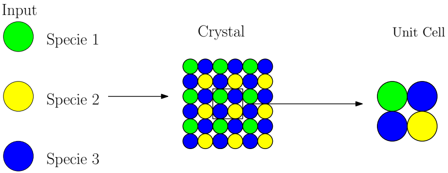
Crystals are everywhere

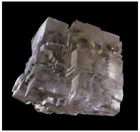


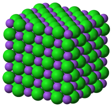
Crystals

Definition (Crystals)

A **Crystal** is a material composed of an (infinitely) repeating **Unit Cell**.



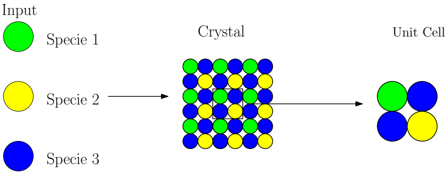


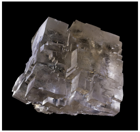


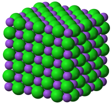
Crystals

Definition (Unit Cells)

A **Unit Cell** is a contiguous region of space containing some set of **Ions**.

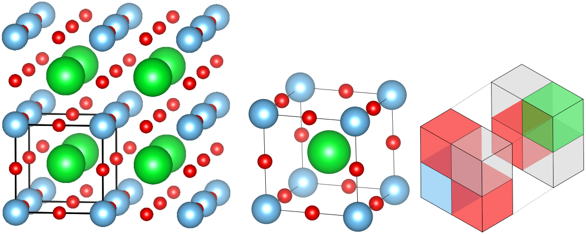






Discrete Crystals

- In this talk we are interested in **Discrete Crystals**, i.e. crystals where every ion is placed on a grid.
- In this model, every cell is either empty, or wholly occupied by an ion.
- For simplicity we assume that each cell can contain only 1 ion, and that each ion can fit into a single cell.



Goals

We want to:

- **Count** the number of potential (discrete) crystal structures from some set of ions and given size.
- **Generate** the complete set of (discrete) crystal structures from some set of ions and given size.
- **Sample** crystals from the space of potential structures with uniform probability.

Discrete Crystals as Multidimensional Words

- Multidimensional words are a natural object for representing discrete crystals.
- Here the alphabet Σ represents the set of ions plus some symbol for empty space.
- The size of the word is the dimensions of the unit cell.

Discrete Crystals as Multidimensional Words

- Multidimensional words are a natural object for representing discrete crystals.
- Here the alphabet Σ represents the set of ions plus some symbol for empty space.
- The size of the word is the dimensions of the unit cell.
- **Advantages**: Multidimensional words are well studied objects, with results on counting, generating, and sampling.

Discrete Crystals as Multidimensional Words

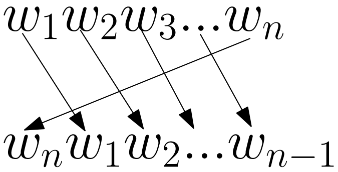
- Multidimensional words are a natural object for representing discrete crystals.
- Here the alphabet Σ represents the set of ions plus some symbol for empty space.
- The size of the word is the dimensions of the unit cell.
- **Advantages**: Multidimensional words are well studied objects, with results on counting, generating, and sampling.
- **Problem**: Crystals have **translational symmetry**.

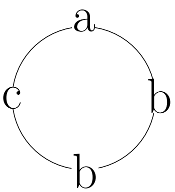
Translational Symmetry

$$\begin{bmatrix} a & a & a \\ a & b & b \\ c & c & b \end{bmatrix}, \quad \begin{bmatrix} a & a & a \\ b & b & a \\ c & b & c \end{bmatrix}, \quad \begin{bmatrix} a & a & a \\ b & a & b \\ b & c & c \end{bmatrix},$$
$$\begin{bmatrix} a & b & b \\ c & c & b \\ a & a & a \end{bmatrix}, \quad \begin{bmatrix} b & b & a \\ c & b & c \\ a & a & a \end{bmatrix}, \quad \begin{bmatrix} b & a & b \\ b & c & c \\ a & a & a \end{bmatrix},$$
$$\begin{bmatrix} c & c & b \\ a & a & a \\ a & b & b \end{bmatrix}, \quad \begin{bmatrix} c & b & c \\ a & a & a \\ b & b & a \end{bmatrix}, \quad \begin{bmatrix} b & c & c \\ a & a & a \\ b & a & b \end{bmatrix}$$

Necklaces

- In 1D the problem of translational symmetry is solved using **Necklaces**.
- Informally a necklace is a set of words that can be reached from each other by some **translation**.
- The **translation** (or cyclic shift) of a word w by some integer i returns the word w' where $w'_j = w_{j-i \bmod n}$.





abbc
bbca
bcab
cabb

Periodicity

- A necklace of length n containing the word w is **periodic** if there is a subword of w that can be repeated to make w.
- A necklace is **aperiodic** if it is not periodic.
- An aperiodic necklace is called a **Lyndon word**.

abab	**aaab**
baba	aaba
abab	abaa
baba	baaa

Some Notation for 1D Necklaces

For the remainder of this talk we use the following assumptions:

- Σ to denotes an alphabet, which we assume has size q.
- The **Canonical form** of a necklace ω (denoted $\langle\omega\rangle$) is the **Lexicographically smallest** word $w \in \omega$.
- $\mathcal{N}_q^n$ denotes the set of necklaces of length n over an alphabet of size q.
- $\mathcal{L}_q^n$ denotes the set of Lyndon words (aperiodic necklaces) in $\mathcal{N}_q^n$.

Results for Necklaces

Necklaces are a well studied object, of note to us are the results on:

- **Counting** the sizes of $\mathcal{N}_q^n$ and $\mathcal{L}_q^n$.
- **Generating** the sets $\mathcal{N}_q^n$ and $\mathcal{L}_q^n$ in order.
- **Ranking** a necklace within the sets $\mathcal{N}_q^n$ and $\mathcal{L}_q^n$.
- **Unranking** a necklace from the sets $\mathcal{N}_q^n$ and $\mathcal{L}_q^n$.

Counting Necklaces and Lyndon words

- The most fundamental result for both necklaces and Lyndon words are the closed from formulas for counting.
- These results follow from the Pólya enumeration and the Möbius inversion formula respectively.

$$\begin{aligned}
|\mathcal{N}_q^n| &= \sum_{d|n} |\mathcal{L}_q^d| &&= \frac{1}{n} \sum_{d|n} \phi\left(\frac{n}{d}\right) q^d \\
|\mathcal{L}_q^n| &= \sum_{d|n} \mu\left(\frac{n}{d}\right) |\mathcal{N}_q^d| &&= \frac{1}{n} \sum_{d|n} \mu\left(\frac{n}{d}\right) q^d
\end{aligned}$$

Where:

- $\phi(x)$ is Euler's totient function.
- $\mu(x)$ is the Möbius function.

Generating Necklaces and Lyndon words

- There have been several algorithms for generating both the sets of necklaces and Lyndon words.
- Fredricksen and Kessler[1] provided an algorithm to generate $\mathcal{N}_q^n$ in $O(n)$ time (proven by Ruskey et al.[2]).
- Catel et. al.[3] provided an $O(n)$ time algorithm to generate only $\mathcal{L}_q^n$.

[1] Harold Fredricksen and Irving J Kessler. "An algorithm for generating necklaces of beads in two colors". In: *Discrete mathematics* 61.2-3 (1986), pp. 181–188.

[2] F. Ruskey, C. Savage, and T. Min Yih Wang. "Generating necklaces". In: *Journal of Algorithms* 13.3 (1992), pp. 414–430. ISSN: 01966774.

[3] K. Cattell et al. "Fast Algorithms to Generate Necklaces, Unlabeled Necklaces, and Irreducible Polynomials over GF(2)". In: *Journal of Algorithms* 37.2 (2000), pp. 267–282.

Ranking necklaces

Definition (Rank)

The **Rank** of a necklace ω in the set $\mathcal{N}_q^n$ is the number of necklaces with a canonical form that is less than or equal to the canonical form of ω.

1.	aaaaaa	2.	aaaaab	3.	aaaabb	4.	aaabab
5.	aaabbb	6.	aabaab	7.	aababb	**8.**	**aabbab**
9.	aabbbb	10.	ababab	11.	ababbb	12.	abbabb
13.	abbbbb	14.	bbbbbb				

Example of the set $\mathcal{N}_2^6$. In bold, the necklace represented by the word aabbab with a rank of 8.

Ranking Necklaces

- The problem of ranking necklaces originates from the problem of ranking de Bruijn Sequences[4].
- The first class of necklaces to be ranked was Lyndon words.
- This was generalised to ranking general necklaces[5],[6] in $O(n^2)$ time.

---

[4] T. Kociumaka, J. Radoszewski, and W. Rytter. "Computing k-th Lyndon word and decoding lexicographically minimal de Bruijn sequence". In: *Symposium on Combinatorial Pattern Matching*. Springer International Publishing, 2014, pp. 202–211.

[5] S. Kopparty, M. Kumar, and M. Saks. "Efficient indexing of necklaces and irreducible polynomials over finite fields". In: *Theory of Computing* 12.1 (2016), pp. 1–27.

[6] J. Sawada and A. Williams. "Practical algorithms to rank necklaces, Lyndon words, and de Bruijn sequences". In: *Journal of Discrete Algorithms* 43 (2017), pp. 95–110. ISSN: 15708667.

Unranking Necklaces

Definition (Uranking Problem)

Given a set of necklaces $\mathcal{N}_q^n$ and an integer i, the unranking problem asks for the necklace with a rank of i.

1.	aaaaaa	2.	aaaaab	3.	aaaabb	4.	aaabab
5.	aaabbb	6.	aabaab	7.	aababb	**8.**	**aabbab**
9.	aabbbb	10.	ababab	11.	ababbb	12.	abbabb
13.	abbbbb	14.	bbbbbb				

Why do we care about ranks?

- Going back to our goal of sampling, note that while randomly sampling words may lead to some necklaces (Lyndon words in particular) being over represented, choosing a rank at random and unranking the corresponding necklaces allows for a uniform distribution.

aaaa	**aaab**	**aabb**	**ab**ab	**abbb**	**b**bbb
-	aaba	abba	baba	bbba	-
-	abaa	bbaa	-	bbab	-
-	baaa	baab	-	babb	-

Example of the set of words of length 4 over the alphabet $\Sigma = \{a, b\}$ *split into the 6 necklaces in* $\mathcal{N}_2^4$. *Highlighted are the periods of the words, corresponding to the number of words in the corresponding necklace.*

Multidimensional Necklaces (almost)

- To define multidimensional necklaces we need three more things:
  1. Notation for multidimensional words.
  2. A formal definition of translational equivalence.
  3. Some notion over ordering over the set of multidimensional words.

Notation for multidimensional words

- We treat d dimensional words as d-dimensional arrays of symbols over some alphabet Σ.
- Given a word w, the notation $w[x_1, x_2, \ldots, x_d]$ is used to denote the symbol at position $x_1, x_2, \ldots, x_d$ in w.
- Given a word w of size $n_1 \times n_2 \times \ldots \times n_d$, the notation w_i is used denote the word v of size $n_1 \times n_2 \times \ldots \times n_{d-1}$ where $v[x_1, x_2, \ldots, x_{d-1}] = w[x_1, x_2, \ldots, x_{d-1}, i]$. We call w_i the i^{th} slice of w.

$$w = \begin{bmatrix} a & a & b \\ a & a & c \\ a & b & c \end{bmatrix}$$

$$w_1 = \begin{bmatrix} a & a & b \end{bmatrix}$$
$$w_2 = \begin{bmatrix} a & a & c \end{bmatrix}$$
$$w_3 = \begin{bmatrix} a & b & c \end{bmatrix}$$

Translational Equivalence

- Let w be a 2D word of size $n_1 \times n_2$ and let (i, j) be a pair of integers such that $0 \leq i \leq n_1 - 1$ and $0 \leq j \leq n_2 - 1$.
- The *translation* of w by (i, j) returns the word v where $w[x, y] = v[x + i \bmod n_1, y + j \bmod n_2]$.
- More generally, given a d-dimensional word w of size $n_1 \times n_2 \times \dots n_d$, the translation by $(t_1, t_2, \dots, t_d)$ returns the word v such that $w[x_1, x_2, \dots, x_d] = v[x_1 + t_1 \bmod n_1, x_2 + t_2 \bmod n_2, \dots, x_d + t_d \bmod n_d]$.
- The notation $w \circ (t_1, t_2, \dots, t_d)$ is used to denote the shift of w by $(t_1, t_2, \dots, t_d)$.

$$\begin{bmatrix} a & a & b \\ a & a & b \\ a & b & c \end{bmatrix} \circ (1, 1) = \begin{bmatrix} a & b & a \\ b & c & a \\ a & b & a \end{bmatrix}$$

Translational Equivalence

- We define the set of translations for words of size $\vec{n} = (n_1, n_2, \ldots, n_d)$, $Z_{\vec{n}}$ by the direct product of the cyclic groups $Z_{n_1} \times Z_{n_2} \times \ldots \times Z_{n_d}$.
- We order the set of translations $Z_{\vec{n}}$ such that $(i, j) < (x, y)$ if and only if $i < x$ or $i = x$ and $j < y$.

$$\begin{bmatrix} (0,0) & (0,1) & (0,2) & (0,3) \\ (1,0) & (1,1) & (1,2) & (1,3) \\ (2,0) & (2,1) & (2,2) & (2,3) \\ (3,0) & (3,1) & (3,2) & (3,3) \end{bmatrix} = \begin{bmatrix} 0\cdot 4+0 & 0\cdot 4+1 & 0\cdot 4+2 & 0\cdot 4+3 \\ 1\cdot 4+0 & 1\cdot 4+1 & 1\cdot 4+2 & 1\cdot 4+3 \\ 2\cdot 4+0 & 2\cdot 4+1 & 2\cdot 4+2 & 2\cdot 4+3 \\ 3\cdot 4+0 & 3\cdot 4+1 & 3\cdot 4+2 & 3\cdot 4+3 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 1 & 2 & 3 \\ 4 & 5 & 6 & 7 \\ 8 & 9 & 10 & 11 \\ 12 & 13 & 14 & 15 \end{bmatrix}$$

Ordering Multidimensional Words

- We order multidimensional words using **Necklace recursive order**.
- This order works by comparing each slice in order.
- Let $\langle w \rangle$ denote the canonical representation of the necklace containing the word w, and let $T(w)$ denote the translation $t \in Z_{\vec{n}}$ such that $\langle w \rangle \circ t = w$.

Definition (Necklace Recursive Order)

Given two words w, v of size $n_1 \times n_2 \times \ldots \times n_d$ over the alphabet Σ, $w \leq v$ if and only if for the smallest i such that $w_i \neq v_i$, either $\langle w_i \rangle < \langle v_i \rangle$ or $\langle w_i \rangle = \langle v_i \rangle$ and $T(w_i) < T(v_i)$.

Ordering Examples

$$w = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} = \begin{bmatrix} a & a & a & b \\ a & a & b & a \\ a & a & a & b \\ b & a & a & a \end{bmatrix}, u = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} a & a & a & b \\ a & a & b & a \\ b & a & a & a \\ b & a & a & a \end{bmatrix},$$

$$v = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} = \begin{bmatrix} a & a & a & b \\ a & a & b & a \\ a & a & b & b \\ b & a & a & a \end{bmatrix}$$

Figure 1: An example of three words, w, u, and v, ordered as follows $w < u < v$. Note that $w_1 : w_2 = v_1 : v_2 = u_1 : u_2$. However, $\langle w_3 \rangle = \langle u_3 \rangle = aaab$, which is smaller than $\langle v_3 \rangle = aabb$. Further, $w_3 < u_3$ as $G(w_3, \langle w_3 \rangle) = 2$ and $G(u_3, \langle u_3 \rangle) = 3$, which is larger than 2.

Multidimensional Necklace

Definition (Multidimensional Necklace)

A **Multidimensional Necklace** ω of size $n_1 \times n_2 \times \ldots \times n_d$ over an alphabet Σ is a set of d-dimensional words of size $n_1 \times n_2 \times \ldots \times n_d$ over an alphabet Σ that are equal under the set of translations $Z_{(n_1, n_2, \ldots, n_d)}$. The set of all such necklaces is denoted $\mathcal{N}_q^{(n_1, n_2, \ldots, n_d)}$.

Definition (Canonical Form)

The **Canonical Form** of a necklace ω, denoted $\langle \omega \rangle$, is the smallest word $w \in \omega$ under the **necklace recursive order**.

Definition (Comparing Necklaces)

Let ω, υ be a pair of necklaces of size $n_1, n_2, \ldots, n_d$ over the alphabet Σ. $\omega < \upsilon$ if and only if $\langle \omega \rangle < \langle \upsilon \rangle$.

Examples of Multidimensional Necklaces

$\begin{bmatrix} a & a \\ a & a \end{bmatrix}$	$\begin{bmatrix} a & a \\ a & b \end{bmatrix}$	$\begin{bmatrix} a & a \\ b & b \end{bmatrix}$	$\begin{bmatrix} a & b \\ a & b \end{bmatrix}$	$\begin{bmatrix} a & b \\ b & a \end{bmatrix}$	$\begin{bmatrix} a & b \\ b & b \end{bmatrix}$	$\begin{bmatrix} b & b \\ b & b \end{bmatrix}$
-	$\begin{bmatrix} a & a \\ b & a \end{bmatrix}$	-	$\begin{bmatrix} b & a \\ b & a \end{bmatrix}$	$\begin{bmatrix} b & a \\ a & b \end{bmatrix}$	$\begin{bmatrix} b & a \\ b & b \end{bmatrix}$	-
-	$\begin{bmatrix} a & b \\ a & a \end{bmatrix}$	$\begin{bmatrix} b & b \\ a & a \end{bmatrix}$	-	-	$\begin{bmatrix} b & b \\ a & b \end{bmatrix}$	-
-	$\begin{bmatrix} b & a \\ a & a \end{bmatrix}$	-	-	-	$\begin{bmatrix} b & b \\ b & a \end{bmatrix}$	-

Aperiodicity in Multidimensional Necklaces

- As in the 1D case, multidimensional necklaces (and words) can be **Aperiodic**.
- A multidimensional necklace ω is **Periodic** if there exists some subword of ω that can be used to tile the space equivalently to ω.
- A multidimensional necklace ω is **Aperiodic** if there exists no such subword.
- An aperiodic multidimensional necklace is called a multidimensional **Lyndon Word**. The set of Lyndon Words of size $n_1 \times n_2 \times \ldots \times n_d$ over an alphabet of size q is denoted $\mathcal{L}_q^{(n_1,n_2,\ldots,n_d)}$.

Aperiodicity in Multidimensional Necklaces

$$\langle \omega \rangle = \begin{bmatrix} \color{blue}{a} & \color{blue}{b} & a & b \\ \color{blue}{b} & \color{blue}{a} & b & a \\ a & b & a & b \\ b & a & b & a \end{bmatrix}, \langle \upsilon \rangle = \begin{bmatrix} a & a & a & b \\ a & a & a & b \\ a & a & a & b \\ a & a & b & a \end{bmatrix}$$

- As in 1D, the number of words in a necklace is bounded from above by the size of the period.

Aperiodicity in Multidimensional Necklaces

$$\langle \omega \rangle = \begin{bmatrix} a & b & a & b \\ b & a & b & a \\ a & b & a & b \\ b & a & b & a \end{bmatrix}, \langle \upsilon \rangle = \begin{bmatrix} a & a & a & b \\ a & a & a & b \\ a & a & a & b \\ a & a & b & a \end{bmatrix}$$

- As in 1D, the number of words in a necklace is bounded from above by the size of the period.
- Unlike in 1D, the number of words may be less than the size of the period.

Atranslational Multidimensional Necklaces

- A necklace of size $n_1 \times n_2 \times \ldots \times n_d$ is **atranslational** if it contains $n_1 \cdot n_2 \cdot \ldots \cdot n_d$ words.
- A word w is atranslational if $w \circ t \neq w \circ r$ for every pair of translations $t, r \in Z_{\vec{n}}$ where $t \neq r$.
- Every atranslational word is aperiodic, however not every aperiodic word is atranslational.
- The set of atranslational necklaces of size $n_1 \times n_2 \times \ldots \times n_d$ over an alphabet of size q is denoted $\mathcal{A}_q^{(n_1, n_2, \ldots, n_d)}$.

Atranslational Multidimensional Necklaces

$$\langle \omega \rangle = \begin{bmatrix} a & a & a & b \\ a & a & b & a \\ a & b & a & a \\ b & a & a & a \end{bmatrix}, \langle \upsilon \rangle = \begin{bmatrix} a & a & a & b \\ a & a & a & b \\ a & a & a & b \\ a & a & b & a \end{bmatrix}$$

- ω is aperiodic, but not atranslational as it only contains 4 words, υ is both aperiodic and atranslational.

Atranslational Multidimensional Necklaces

$\begin{bmatrix} a & a \\ a & b \end{bmatrix}$	$\begin{bmatrix} a & b \\ b & a \end{bmatrix}$	$\begin{bmatrix} a & b \\ b & b \end{bmatrix}$
$\begin{bmatrix} a & a \\ b & a \end{bmatrix}$	$\begin{bmatrix} b & a \\ a & b \end{bmatrix}$	$\begin{bmatrix} b & a \\ b & b \end{bmatrix}$
$\begin{bmatrix} a & b \\ a & a \end{bmatrix}$	$\begin{bmatrix} b & a \\ a & b \end{bmatrix}$	$\begin{bmatrix} b & b \\ a & b \end{bmatrix}$
$\begin{bmatrix} b & a \\ a & a \end{bmatrix}$	$\begin{bmatrix} a & b \\ b & a \end{bmatrix}$	$\begin{bmatrix} b & b \\ b & a \end{bmatrix}$

- Example of the set of aperiodic 2×2 words over a binary alphabet sorted into necklace classes.
- Note that $\begin{bmatrix} a & b \\ b & a \end{bmatrix}$ is aperiodic, as it can be represented using any subword (without applying some translation), but it is not atranslational, as there are only two unique translations.

Results Overview

Let $\vec{n} = (n_1, n_2, \ldots, n_d)$, $N = \prod_{i=1}^{d} n_i$ and q be the size of the alphabet Σ. We obtain the following results for multidimensional necklaces, Lyndon words, and atranslational necklaces of size $n_1 \times n_2 \times \ldots \times n_d$ over Σ:

- Closed from formulae for counting the sizes of $\mathcal{N}_q^{\vec{n}}, \mathcal{L}_q^{\vec{n}}$, and $\mathcal{A}_q^{\vec{n}}$.
- An $O(N)$ amortised time algorithm for generating the set $\mathcal{N}_q^{\vec{n}}$ in Necklace recursive order.
- An $O(N^5)$ time algorithm for ranking necklaces within the sets $\mathcal{N}_q^{\vec{n}}, \mathcal{L}_q^{\vec{n}}$, and $\mathcal{A}_q^{\vec{n}}$.
- An $O(N^{6(d+1)} \cdot \log^d(q))$ time unranking algorithm for the sets $\mathcal{N}_q^{\vec{n}}, \mathcal{L}_q^{\vec{n}}$, and $\mathcal{A}_q^{\vec{n}}$.

Counting Multidimensional Necklaces

- Our counting results are obtained using the Pólya enumeration and the Möbius inversion formulae for necklaces and Lyndon words respectively.

$$|\mathcal{N}_q^{\vec{n}}| = \frac{1}{N} \sum_{f_1 | n_1} \phi(f_1) \sum_{f_2 | n_2} \phi(f_2) \dots \sum_{f_d | n_d} \phi(f_d)\, q^{(N/lcm(f_1, f_2, \dots, f_d))}$$

$$|\mathcal{L}_q^{\vec{n}}| = \sum_{f_1 | n_1} \mu\left(\frac{n_1}{f_1}\right) \sum_{f_2 | n_2} \mu\left(\frac{n_2}{f_2}\right) \dots \sum_{f_d | n_d} \mu\left(\frac{n_d}{f_d}\right) |\mathcal{N}_q^{f_1, f_2 \dots f_d}|$$

Counting Multidimensional Necklaces

- Our counting results are obtained using the Pólya enumeration and the Möbius inversion formulae for necklaces and Lyndon words respectively.

$$|\mathcal{N}_q^{\vec{n}}| = \frac{1}{N} \sum_{f_1 | n_1} \phi(f_1) \sum_{f_2 | n_2} \phi(f_2) \dots \sum_{f_d | n_d} \phi(f_d)\, q^{(N/lcm(f_1, f_2, \dots, f_d))}$$

$$|\mathcal{L}_q^{\vec{n}}| = \sum_{f_1 | n_1} \mu\left(\frac{n_1}{f_1}\right) \sum_{f_2 | n_2} \mu\left(\frac{n_2}{f_2}\right) \dots \sum_{f_d | n_d} \mu\left(\frac{n_d}{f_d}\right) |\mathcal{N}_q^{f_1, f_2 \dots f_d}|$$

- Counting atranslational words is more complicated.

Intuition behind counting Atranslational Necklaces

- At a (very) high level, the idea behind counting atranslational necklaces is to determine the number of **translational, aperiodic necklaces**.
- This is done in a recursive manner, noting that every translational, aperiodic necklace is composed of some subword that has been repeated under a translation.
- By counting the number of such words, we are able to determine the number of aperiodic necklaces.

$$w = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} = \begin{bmatrix} a & a & a & b \\ a & a & b & a \\ a & b & a & a \\ b & a & a & a \end{bmatrix} = \begin{bmatrix} w_1 \circ (0) \\ w_1 \circ (1) \\ w_1 \circ (2) \\ w_1 \circ (3) \end{bmatrix}$$

Counting Atranslational Necklaces

$$|L_q^{\vec{n}}| - \sum_{i \in [d]} \sum_{l | n_i} \begin{cases} 0 & l = n_i \\ \left(\prod_{t=i+1}^{d-1} -\mu(n_t) \right) \left(-\mu\left(\frac{n_i}{l}\right)\right) |\mathcal{A}_q^{n_1, n_2, \ldots, n_{d-1}, l}| \cdot H(i, l, \vec{n}, d) & 1 < l < n_d \end{cases}$$

Where:

- $H(i, l, \vec{n}, d) = \prod_{j=i}^{d} \begin{cases} 1 & i = d \\ (|\mathbf{G}(1, \vec{n})| - (I(i, l, \vec{n}))) \cdot (H(i, l, (n_1, n_2, \ldots, n_{d-1}), d-1)) & i < d \end{cases}$
- $I(i, l, (n_1, n_2, \ldots, n_d)) = \begin{cases} 0 & i = d \text{ or } l > 1 \\ 1 + I(i, l, (n_1, n_2, \ldots, n_{d-1})) & n_i = n_d \\ I(i, l, (n_1, n_2, \ldots, n_{d-1})) & n_i \neq n_d \end{cases}$

Generating Multidimensional Necklaces

- At a high level our algorithm works by generating the set of **Prenecklaces** (prefixes of necklaces) in order.
- Our generation algorithm relies on the Necklace Recursive Ordering as a means to do so.
- We show a worked example of how to generate the necklace in $\mathcal{N}_2^{(4,4)}$ following $\begin{bmatrix} a & a & a & a \\ a & a & a & b \\ a & b & a & a \\ b & b & b & b \end{bmatrix}$.

Prenecklaces

Definition (Prefix)

A word w of size $n_1 \times n_2 \times \ldots \times n_d$ is a **prefix** of the word v of size $n_1 \times n_2 \times \ldots \times n_{d-1} \times m$ if $m \geq n_d$ and $w_i = v_i$ for every $i \in 1, 2, \ldots, n_d$.

Definition (Prenecklaces)

A word w of size $n_1 \times n_2 \times \ldots \times n_d$ is a **prenecklace** if there exists a word v of size $n_1 \times n_2 \times \ldots \times n_{d-1} \times m$ such that $v = \langle v \rangle$ and w is a prefix of v.

Necklace Generation Example

$$\begin{bmatrix} a & a & a & a \\ a & a & a & b \\ a & b & a & a \\ b & b & b & b \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ a & a & a & a \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ a & a & a & b \end{bmatrix}$$

Necklace Generation Example

$$\begin{bmatrix} a & a & a & a \\ a & a & a & b \\ a & b & a & a \\ \mathbf{b} & \mathbf{b} & \mathbf{b} & \mathbf{b} \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ a & a & a & a \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ a & a & a & b \end{bmatrix}$$

Necklace Generation Example

$$\begin{bmatrix} a & a & a & a \\ a & a & a & b \\ a & b & a & a \\ \mathbf{b} & \mathbf{b} & \mathbf{b} & \mathbf{b} \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ a & a & a & a \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ a & a & a & b \end{bmatrix}$$

Necklace Generation Example

$$\begin{bmatrix} a & a & a & a \\ a & a & a & b \\ a & b & a & a \\ b & b & b & b \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ a & a & a & a \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ a & a & a & b \end{bmatrix}$$

Necklace Generation Example

$$\begin{bmatrix} a & a & a & a \\ a & a & a & b \\ a & b & a & a \\ b & b & b & b \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ a & a & a & a \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ a & a & a & b \end{bmatrix}$$

Necklace Generation Example

$$\begin{bmatrix} a & a & a & a \\ a & a & a & b \\ a & b & a & a \\ b & b & b & b \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{a} & \mathbf{a} & \mathbf{a} & \mathbf{a} \\ a & a & a & b \\ b & a & a & a \\ \mathbf{a} & \mathbf{a} & \mathbf{a} & \mathbf{a} \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ a & a & a & b \end{bmatrix}$$

Necklace Generation Example

$$\begin{bmatrix} a & a & a & a \\ a & a & a & b \\ a & b & a & a \\ b & b & b & b \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ a & a & a & a \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ a & a & a & b \end{bmatrix}$$

Necklace Generation Example

$$\begin{bmatrix} a & a & a & a \\ a & a & a & b \\ a & b & a & a \\ b & b & b & b \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ \mathbf{a} & \mathbf{a} & \mathbf{a} & \mathbf{a} \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ \mathbf{a} & \mathbf{a} & \mathbf{a} & \mathbf{b} \end{bmatrix}$$

Necklace Generation Example

$$\begin{bmatrix} a & a & a & a \\ a & a & a & b \\ a & b & a & a \\ b & b & b & b \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ a & a & a & a \end{bmatrix} \rightarrow \begin{bmatrix} a & a & a & a \\ a & a & a & b \\ b & a & a & a \\ a & a & a & b \end{bmatrix}$$

Ranking

- Let ω be the necklace we are ranking and $\mathcal{N}_q^{\vec{n}}$ be the set we are ranking ω in.
- The high level idea behind our ranking algorithm is to compute the number of words belonging to a necklace class that is smaller than $\langle \omega \rangle$.
- This is transformed first into the number of Atranslational necklaces smaller than ω, then into the number of Lyndon words smaller than ω.
- Our recursive ordering allows this process to be handled recursively.

Unranking

- Let i be the rank of the necklace in $\mathcal{N}_q^{\vec{n}}$ and let $\omega, \upsilon \in \mathcal{N}_q^{\vec{n}}$ be a pair of necklaces such that:
  - $\omega \leq \upsilon$.
  - w is the longest prefix that is shared by both $\langle \omega \rangle$ and $\langle \upsilon \rangle$.
  - ω is the smallest necklace such that w is a prefix of $\langle \omega \rangle$.
  - υ is the largest necklace such that w is a prefix of $\langle \upsilon \rangle$.
- Then the number of necklaces sharing w as a prefix is given by $Rank(\upsilon) - Rank(\omega)$ and further the necklace with a rank of i has w as a prefix if and only if $Rank(\omega) \leq i \leq Rank(\upsilon)$.
- This can be used with a binary search to determine the i^{th} necklace by determining the prefix iteratively.

Open Problems

- Can de Bruijn Tori be found efficiently?
- Can multidimensional necklaces be ranked in $O(N^2)$.
- Can our results be extended to other notions of symmetry?