

CDM

Actions and Counting

Klaus Sutner
Carnegie Mellon University

Fall 2011

Outline:

Outline

- 1 Some Counting Problems
- 2 Actions
- 3 Burnside's lemma
- 4 Some Applications

Some Counting Problems:

Counting, with a Twist

Counting problems are a staple of combinatorics: given some space X , determine its cardinality.

We will here consider a variant: we have an equivalence relation on X , and we want to count the equivalence classes (patterns) rather than the individual objects.

There are lots of examples where it is natural to identify objects: for example, we may want to study geometric figures up to rotational symmetry. Alas, counting patterns rather than individual objects is often a bit harder.

Here are some examples.

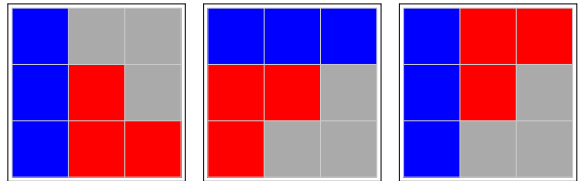
Some Counting Problems:

Problem 1: Tic-Tac-Toe

How many different ways are there to place 3 crosses and 3 noughts on a (standard 3×3) Tic-Tac-Toe board?

No problem: there are $\binom{9}{3,3,3} = 1680$ possible placements.

True, but what if we wanted to identify boards that can be obtained from rotation, or looking through a mirror?



Some Counting Problems:

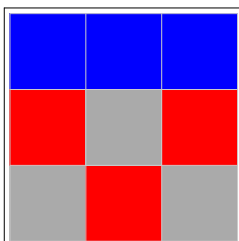
Eyeballing It

There are 4 rotations, and 4 reflections for the Tic-Tac-Toe board.

In all, a single placement of marks can have as many as 8 variants.

So we should expect some $1680/8 = 210$ patterns – placements “up to equivalence”.

But that is not quite right: some placements have less than 8 variants, so we are under-counting the number of patterns.



Some Counting Problems:

Problem 2: Boolean Circuits

Here is a similar but somewhat less frivolous problem. Suppose we want to implement Boolean functions $f : \mathbb{B}^n \rightarrow \mathbb{B}$ as circuits.

There are 2^{2^n} such functions, but we don't need as many circuits. For example, we may have $f(x, y, z) = g(y, z, x)$ so it suffices to implement either f or g .

In general, we can permute the variables arbitrarily: two functions f and g are equivalent if

$$f(\mathbf{x}) = g(\pi(\mathbf{x}))$$

for some permutation π .

How many Boolean functions are there modulo input permutations?

Another Version

Permuting the inputs is an obvious modification of a circuit, but there are other possibilities.

For example, it is straightforward to negate input bits and/or the output bit. Thus, there is another equivalence

$$f(x) = g(x \oplus c) \oplus d$$

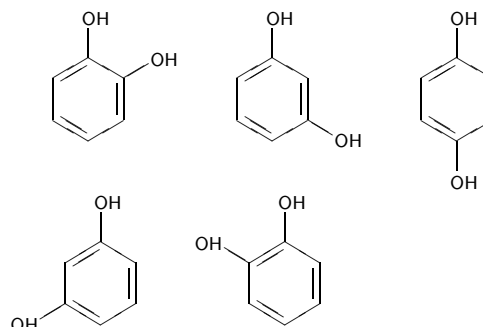
where $c \in \mathbb{2}^n$, $d \in \mathbb{2}$ and \oplus denotes exclusive or.

And, of course, we can combine permutations and negations.

So how many functions $\mathbb{B}^n \rightarrow \mathbb{B}$ are there in all these cases?

Problem 3: Chemistry

Combinatorial (or computational) chemistry leads to many counting problems. For example, suppose we want to enumerate carbocycles like benzene, where some H atoms have been replaced by OH groups.



Burnside, Pólya and Redfield

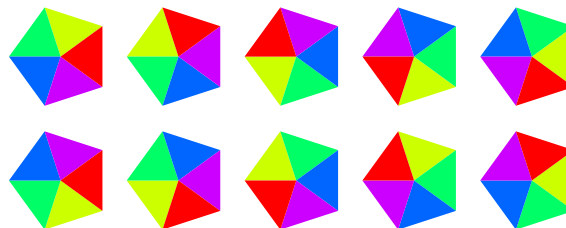
To tackle this type of problem in style one can use some ideas from classical algebra:

- groups
- subgroups
- homomorphisms
- actions

These algebraic ideas are all 19th century, but the applications are 20th century.

Central Idea

Groups describe symmetries, so if we want to count modulo symmetry it is natural to use groups to do so.



Counting Problems

Recall the Tic-Tac-Toe problem from last time: we want to count boards modulo equivalence, where two boards are considered equivalent if one can be moved to the other by rotations and/or reflections.

More precisely, we have to consider the interaction between Tic-Tac-Toe boards and the elements of the dihedral group D_4 .

We can safely assume that the group in question is always a subgroup of a symmetric group, so we are dealing with a class of permutations.

What is needed is some glue that connects the permutations with the objects we are interested in (such as the boards, or carbocycles, or circuits, ...)

Moving Things Around

To tackle this issue we need to explain how permutations "rearrange" objects.

Consider a list of n objects:

$$x = (x_1, x_2, \dots, x_n)$$

We can use any permutation f in \mathbb{S}_n to rearrange the elements of X :

$$x' = (x_{f(1)}, x_{f(2)}, \dots, x_{f(n)})$$

This is fairly clear intuitively, at least when the group has some nice geometric meaning. Still, it's a good idea to give a precise definition so we can handle more general cases and solve a variety of counting problems.

Permutations Acting on Objects

We are dealing with

- a collection X of objects (configurations),
- a group G of permutations.

Given a group element a and an object x we get a new object x' by letting “ a act on x ”.

The crucial condition: the action has to coexist peacefully with the group operation. For example, we want $x = x'$ for $a = 1$. And a^{-1} acting on x' should produce x .

We need to axiomatize these properties.

Left Actions

Definition

Let G be a group and X a set. A **left action of G on X** is a function φ such that

$$\begin{aligned}\varphi : G \times X &\rightarrow X \\ \varphi(a * b, x) &= \varphi(a, \varphi(b, x)) \\ \varphi(1, x) &= x\end{aligned}$$

X is also called a **G -set**.

Notation:

It is customary to write $a \cdot x$ or even ax instead of $\varphi(a, x)$. Hence

$$\begin{aligned}(a * b) \cdot x &= a \cdot (b \cdot x) \\ 1 \cdot x &= x\end{aligned}$$

This is much better notation, albeit slightly dangerous.

Example: Orbits

Suppose we have a bijection $f : A \rightarrow A$ on some ground set. Then the (additive) group \mathbb{Z} naturally acts on A on the left via

$$k \cdot x = f^k(x)$$

This works since

$$\begin{aligned}\varphi(0, x) &= f^0(x) = x \\ \varphi(k + l, x) &= f^{k+l}(x) = f^k(f^l(x)) = \varphi(k, \varphi(l, x))\end{aligned}$$

Example: Monoid Actions

Note that the definition of an action does not actually require G to be a group, a monoid is enough.

This generalization is often very convenient.

For example, we can get the monoid \mathbb{N} acting on A for any arbitrary function $f : A \rightarrow A$ (we don't need to have a bijection to go backwards).

The orbit of a point x is then just

$$\{k \cdot x \mid x \in \mathbb{N}\}.$$

Incidentally, this also makes sense in continuous time where we have \mathbb{R}_0^+ acting on A .

The Standard Example

Claim

The operation $f \cdot x = (x_{f(1)}, x_{f(2)}, \dots, x_{f(n)})$ is a left action.

Proof.

It is clear that $1 \cdot x = x$.

Consider $(f \circ g) \cdot x = y$ versus $f \cdot (g \cdot x) = z$. We need to show that $y = z$.

Recall that we compose functions from left to right, so that $y = (x_{g(f(1))}, \dots, x_{g(f(n))})$.

But then $z = f \cdot (x_{g(1)}, \dots, x_{g(n)}) = y$.

□

Right?

Stop! Mental Health Alert

Think carefully – this looks absolutely wrong, but it's right. Take a good look at the following. Write $u_i = x_{g(i)}$.

$$\begin{aligned}g \cdot x &= (x_{g(1)}, x_{g(2)}, \dots, x_{g(n)}) \\ &= (u_1, u_2, \dots, u_n) \\ f \cdot u &= (u_{f(1)}, u_{f(2)}, \dots, u_{f(n)}) \\ &= (x_{g(f(1))}, x_{g(f(2))}, \dots, x_{g(f(n))})\end{aligned}$$

Yes?

And Right Actions

As you might have suspected all along: Where there's a left, there must be a right ...

Definition

Let G be a group and X a set. A **right action of G on X** is a function φ such that

$$\begin{aligned} \varphi : X \times G &\rightarrow X \\ \varphi(x, a * b) &= \varphi(\varphi(x, a), b) \\ \varphi(x, 1) &= x \end{aligned}$$

Needless to say, this is often written $x \cdot a$ and xa .

To maintain sanity, we always write a, b, c, \dots for group elements and x, y, z, \dots for the elements of X .

Nil novis sub solem

If right actions look familiar, it's because we have seen this idea already: deterministic finite state machines also use actions. In fact, we can think of transition function in a DFA as a (monoid) right action where the monoid in question is the collection of all words over the input alphabet of the automaton.

$$\begin{aligned} \delta : Q \times \Sigma^* &\rightarrow Q \\ \delta(p, \varepsilon) &= p \\ \delta(p, xy) &= \delta(\delta(p, x), y) \end{aligned}$$

The reason a DFA is a natural example of a right action is that we are scanning words from left to right. If we were reading right to left we would get a left action instead.

YARA

Yet Another Right Action.

Recall the operation of prefix quotients on languages: $w^{-1}L$ is obtained by deleting prefix w from words in L .

$$w^{-1}L = \{v \mid wv \in L\}$$

Since the prefix is deleted at the beginning (left end) of the word this is usually written on the left, as indicated – but these quotients are another example of a right action of Σ^* (this time acting on the collection of all languages).

Since we write on the left we get the awkward inversion in $(uv)^{-1}L = v^{-1}u^{-1}L$.

Exercises

Exercise

Make sure you really understand the difference between left and right actions. What would happen if we did functional composition the other way around?

Exercise

Check carefully that the last two examples really are right actions of the monoid Σ^* . Find better notation for the quotient operation.

Exercise

Find some more natural examples for left and right actions, for groups as well as monoids.

Left versus Right

Let's come back to our primary example of a group action: some permutation f from the symmetric group \mathbb{S}_n permuting the elements of $x = (x_1, x_2, \dots, x_n)$.

Let $f(i) = j$.

type	intuitively	result
left	" x_i is replaced by x_j "	$(x_{f(1)}, \dots, x_{f(n)})$
right	" x_i moves to x_j "	$(x_{f^{-1}(1)}, \dots, x_{f^{-1}(n)})$

Having two versions to deal with might seem plain annoying, but there are occasions when left actions are more convenient to work with, and there are occasions when right actions are more convenient. Grin and bear it.

Left is Right

For group actions we can interchange left and right in the following sense.

Proposition

Consider two maps $\varphi : G \times X \rightarrow X$ and $\psi : X \times G \rightarrow X$ such that $\psi(x, a) = \varphi(a^{-1}, x)$. Then φ is a left action if, and only if, ψ is a right action.

Proof. Suppose φ is a left action.

$$\begin{aligned} \psi(x, a * b) &= \varphi((a * b)^{-1}, x) \\ &= \varphi(b^{-1} * a^{-1}, x) \\ &= \varphi(b^{-1}, \varphi(a^{-1}, x)) \\ &= \psi(\psi(x, a), b) \end{aligned}$$

The other direction is entirely similar. □

Right is Left

Another way to establish a connection between left and right actions is reverse the multiplication. Given a group $\mathcal{G} = \langle G, \cdot \rangle$ define a new group

$$\mathcal{G}^{\text{op}} = \langle G, * \rangle \quad a * b = b \cdot a.$$

It is not hard to check that \mathcal{G}^{op} is in fact a group.

Now any left action φ over \mathcal{G} translates into a right action ψ over \mathcal{G}^{op} by

$$\psi(x, a) = \varphi(a, x)$$

Exercise

Give a detailed proof of this claim.

Mathematics versus Implementation

From a sufficiently abstract perspective, left and right actions are the same: it doesn't matter much if we replace each group element by its inverse or change the order of multiplication.

In fact, there are older texts that just speak about "a group acting on a set". E.g.,

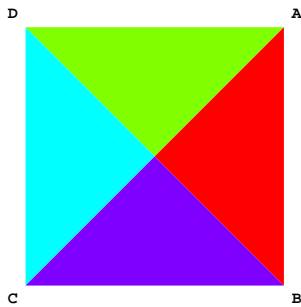
N. G. de Bruijn, Pólyas theory of counting (1964)

has an excellent introduction to the subject without ever mentioning left versus right actions.

That's fine, but when it comes to actual implementation one has to be more careful, the code for both versions is different. More importantly, you must never mix the two versions within the same algorithm.

Dihedral Groups

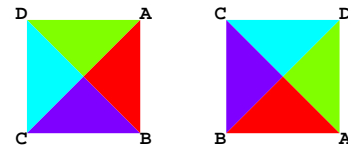
Consider the square with four vertices $X = \{A, B, C, D\}$ in positions $(\pm 1, \pm 1)$.



Dihedral Groups and Permutations

Every permutation σ in \mathbb{S}_4 rearranges the vertices.

For example, acting on the right, permutation $\sigma = T(2, 3, 4, 1)$ causes the clockwise rotation $A \rightarrow B \rightarrow C \rightarrow D \rightarrow A$.



Note that not all permutations make geometric sense: $T(1, 2, 4, 3)$ does not correspond to a rigid motion of the plane that leaves the square in the same place.

The motions allowed in this case are rotations around the origin and reflections along the axes and diagonals.

Selecting a Subgroup

The permutations in \mathbb{S}_4 that make geometric sense are

$$H = \{ T(1, 2, 3, 4), T(2, 3, 4, 1), T(3, 4, 1, 2), T(4, 1, 2, 3), \\ T(2, 1, 4, 3), T(1, 4, 3, 2), T(4, 3, 2, 1), T(3, 2, 1, 4) \}$$

Hence we have a subgroup of $H \subseteq \mathbb{S}_4$ isomorphic to the dihedral group D_4 . It is crucial here to represent D_4 as a subgroup of a permutation group, as an abstract group we would not know how to express its action on $\{A, B, C, D\}$.

We know already that H can be generated by two elements (a rotation and a reflection)

$$H = \langle (1, 2, 3, 4), (1, 2)(3, 4) \rangle$$

Patterns and Orbits

We can now describe patterns by having the whole group act on an element in X : "equivalent" objects are related via the action of some group element.

Definition

Let $\varphi : G \times X \rightarrow X$ be a left action. The **orbit** of $x \in X$ under G is

$$Gx = \{ ax \mid a \in G \}.$$

One says that the elements in an orbit are **G-equivalent**.

These orbits are much more general than our old orbits obtained by iterating a function $f : A \rightarrow A$.

The latter correspond to just the additive monoid \mathbb{N} acting on A .

Chopping Things Up

Proposition

Let G be a group. Then the orbits Gx form a partition of X .

Proof. $z \in Gx \cap Gy$ implies $ax = z = by$ for some $a, b \in G$. But G is a group, so $x = (a^{-1}b)y \in Gy$. \square

So, our terminology makes sense: the blocks of this partition are exactly the patterns we are interested in.

Note, though, that we really need G to be a group, the argument fails for monoids. Over a monoid, all we have is a basin of attraction.

- Some Counting Problems
- Actions
- ③ Burnside's lemma
- Some Applications

Stabilizers and Invariants

Now comes the important idea: using subgroups and fixed points to count.

Definition

Let $\varphi : G \times X \rightarrow X$ be a group left action.

The **stabilizer** of G at $x \in X$ is

$$G_x = \{ a \in G \mid ax = x \}$$

The **invariant subset** of X at $a \in G$ is

$$X_a = \{ x \in X \mid ax = x \}$$

So in a sense we are talking about fixed points here, once from the perspective of the group, and once from the perspective of the G -set.

How to Count

The idea is this: we want to determine the size of Gx .

An upper bound is $|G|$, but usually this bound is not tight. The problem is that we may have $ax = bx$. But note

$$ax = bx \iff a^{-1}bx = x \iff a^{-1}b \in G_x \iff b \in aG_x$$

It will turn out that G_x is a subgroup of G , so we have

$$|Gx| = |\{ ax \mid a \in G \}| = [G : G_x]$$

Stabilizers

Proposition

The stabilizers G_x are subgroups of G .

Proof.

Let $a, b \in G_x$.

Then

$$(a^{-1}b)x = a^{-1}(bx) = a^{-1}(x) = a^{-1}(ax) = x$$

Hence $a^{-1}b \in G_x$ and we are done. \square

Orbit Sizes

Proposition

The index $[G : G_x]$ is the size of the orbit of x .

Proof.

We already know that

$$ax = bx \iff b \in aG_x$$

Hence $|G_x|$ many elements in G produce the same element in the orbit.

So $|Gx| = [G : G_x]$. \square

Enumeration

Suppose we have a coset transversal $\{g_1, g_2, \dots, g_r\}$ where $r = [G : G_x]$. In other words,

$$G/G_x = \{g_1 G_x, g_2 G_x, \dots, g_r G_x\}$$

Then any G -orbit has the form

$$Gx = \{g_1 x, g_2 x, \dots, g_r x\}$$

Hence, if we know representatives for the cosets then we can enumerate the orbit of x directly without repetitions.

The bad news: We can always choose $g_1 = 1$, but other than that it may not be so easy to get at the g_i .

Digression: Double Counting

Double counting is a very simple but sometimes surprisingly powerful idea.

Suppose R (for rows) and C (for columns) are two finite sets and M is an R by C matrix with 0/1 entries.

Let $\text{row}(r)$ be the number of 1's in row $r \in R$, and let $\text{col}(c)$ be the number of 1's in columns $c \in C$.

Then

$$\sum_{r \in R} \text{row}(r) = \sum_{c \in C} \text{col}(c).$$

Yes, yes, that's trivial. But ...

Application

Lemma

$$\sum_{a \in G} |X_a| = \sum_{x \in X} |G_x|.$$

Proof.

Consider the **action matrix**: a G by X matrix M defined by

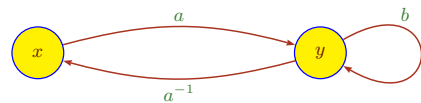
$$M(a, x) = \begin{cases} 1 & \text{if } ax = x, \\ 0 & \text{otherwise.} \end{cases}$$

The rows are bitvectors for the invariant sets, and the columns are bitvectors for the stabilizers. □

Orbits and Conjugacy

What is the relationship between the stabilizers of elements of the same orbit?

Say, $ax = y$. Then $by = y$ iff $a^{-1}ba x = x$.



Thus, G_x and G_y are **conjugate** subgroups: $G_y = aG_x a^{-1}$.

In particular if the group G is commutative, all the stabilizers along a single orbit are the same.

Burnside's lemma

Theorem

Let N be the number of distinct orbits of G acting on X . Then

$$N = \frac{1}{|G|} \sum_{a \in G} |X_a|.$$

Proof.

$$\frac{1}{|G|} \sum_{a \in G} |X_a| = \frac{1}{|G|} \sum_{x \in X} |G_x| = \sum_{x \in X} \frac{1}{[G : G_x]} = \sum_{x \in X} \frac{1}{|G_x|} = N. \quad \square$$

Computing Burnside

In practice, this means that one has to

- Determine the group of actions G . Determine here means having a clear understanding of what the group elements are and how to enumerate them.
- Compute the (sizes of the) invariant sets X_a for all group elements a .

Clearly this can be problematic when the group is large, or when the action is very complicated. A computer algebra system can be very helpful.

- Some Counting Problems
- Actions
- Burnside's lemma
- Some Applications

Warm-Up: Flipping Bits

Here is a trivial counting problem, but it provides an opportunity to use the new machinery.

We consider binary lists of length n .

We want to identify two lists when one is obtained from the other by flipping all bits.

To apply Burnside let

$$X = 2^n$$

$$G = \{1, s\}$$

where s means flip all bits. Hence $s^2 = 1$ and G really is a group.

In fact, G is isomorphic to \mathbb{Z}_2 but let's use the multiplicative notation.

Compute Invariants

We need to calculate the invariant sets, which is easy in this case.

$$X_1 = X$$

$$X_s = \emptyset$$

Hence

$$N = \frac{1}{2}(2^n + 0) = 2^{n-1}$$

So each orbit has the form $\{x, sx\}$ and has size 2.

Not very exciting, but at least it's correct.

Application 1: Cellular Automata

A condensed version of the problem: a cellular automaton is represented by a binary list of length n .

Two lists L and K are equivalent if K can be obtained from L by any combination of reversing the list, or flipping all bits.

To apply Burnside let

$$X = 2^n$$

$$G = \langle r, s \mid r^2 = s^2 = 1, rs = sr \rangle = \{1, r, s, rs\}$$

where r means reversal, s means flip all bits. Note that G is isomorphic to $\mathbb{Z}_2 \times \mathbb{Z}_2$, the Kleinsche Vierergruppe.

The Invariants

Suppose n is even (automatically true in CA application).

$$X_1 = X$$

$$X_r = \{uu^{op} \mid u \in 2^{n/2}\}$$

$$X_s = \emptyset$$

$$X_{rs} = \{u\bar{u}^{op} \mid u \in 2^{n/2}\}$$

Hence

$$N = \frac{1}{4}(2^n + 2^{n/2} + 0 + 2^{n/2}) = 2^{n-2} + 2^{n/2-1}$$

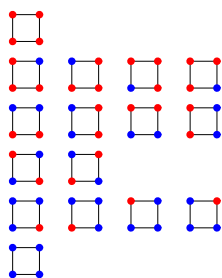
Application 2: Coloring a Square

Color the corners of a square red and blue. Obviously there are 2^4 colorings (our configurations).

Now suppose we do not wish to distinguish between colorings that can be obtained from each other by rotations and reflections (our patterns).

For 2 colors and 4 vertices we can easily compute this to death, but think about the analogous problem with c colors and n vertices.

Brute Force



Enumerate all configurations and group them by hand.

Applying Burnside

So there are 6 patterns.

Let X be the set of all colorings. It is convenient to think of X as 2^4 (read off the colors in clockwise order).

The patterns are exactly the orbits of $x \in X$ under D_4 .

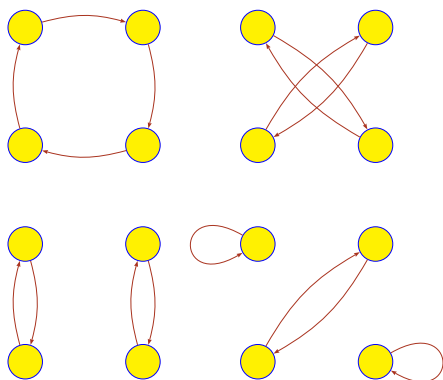
Hence

$$N = \frac{1}{8} \sum_{a \in D_4} |X_a|.$$

So what are the invariant sets?

Recall $D_4 = \{1, \alpha, \alpha^2, \alpha^3, \beta, \alpha\beta, \alpha^2\beta, \alpha^3\beta\}$

Following the Points



The Invariants

$$X_1 = X$$

$$X_\alpha = X_{\alpha^3} = \{0000, 1111\}$$

$$X_{\alpha^2} = \{0000, 0101, 1010, 1111\}$$

$$X_\beta = \{0000, 0011, 1100, 1111\}$$

$$X_{\alpha\beta} = \{i j i k \mid i, j, k \in \mathbf{2}\}$$

$$X_{\alpha^2\beta} = \{0000, 0101, 1010, 1111\}$$

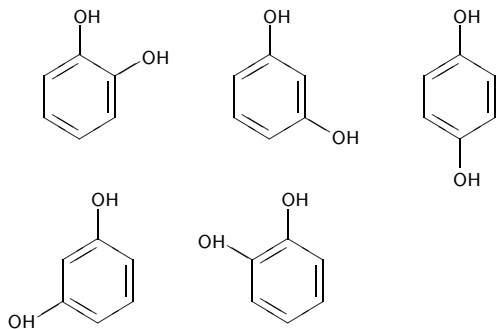
$$X_{\alpha^3\beta} = \{j i k i \mid i, j, k \in \mathbf{2}\}$$

Hence

$$N = \frac{1}{8} (16 + 2 + 4 + 2 + 4 + 8 + 4 + 8) = 6$$

Application 3: Chemistry

Combinatorial (or computational) chemistry leads to many counting problems. For example, suppose we want to enumerate carbocycles like benzene, where some H atoms have been replaced by OH groups.



No good, we need to identify some of these compounds.

The Model: Bracelets

A k -ary bracelet is a circular string of beads in k different colors: do not distinguish between variants obtained by rotation or reflection.

It is customary to represent each equivalence class by its lexicographically first element.

Example: all 21 ternary bracelets of length 4.

(1, 1, 1, 1) (1, 1, 1, 2) (1, 1, 1, 3) (1, 1, 2, 2)
 (1, 1, 2, 3) (1, 1, 3, 3) (1, 2, 1, 2) (1, 2, 1, 3)
 (1, 2, 2, 2) (1, 2, 2, 3) (1, 2, 3, 2) (1, 2, 3, 3)
 (1, 3, 1, 3) (1, 3, 2, 3) (1, 3, 3, 3) (2, 2, 2, 2)
 (2, 2, 2, 3) (2, 2, 3, 3) (2, 3, 2, 3) (2, 3, 3, 3)
 (3, 3, 3, 3)

Setting Things Up

To apply Burnside note that the group acting on X is the same as for a regular n -gon, so we can use

$$X = [k]^n$$

$$G = D_n$$

What are the invariant sets?

We need the cardinality of

$$X_{\alpha^p} = \{x \in X \mid \alpha^p x = x\}$$

$$X_{\alpha^p \beta} = \{x \in X \mid \alpha^p \beta x = x\}$$

For the rotations this means $x_0 = x_p = x_{2p} = \dots$ and things wrap around modulo n .

Déjà Vu All Over Again

Remember the function

$$f_p : \mathbb{Z}_n \rightarrow \mathbb{Z}_n$$

$$z \mapsto z + p \pmod n$$

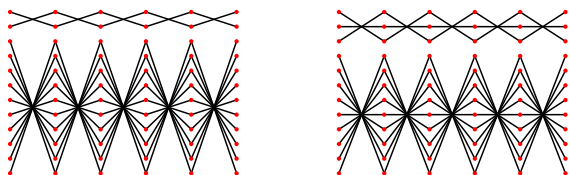
f_p has $\gcd(n, p)$ distinct orbits, each of length $n/\gcd(n, p)$.

$\alpha^p x = x$ means that the list elements on each orbit of f_p are the same.

Hence there are $k^{\gcd(n, p)}$ invariant lists for the rotation α^p .

And Reflections?

How about the reflections $\alpha^k \beta$? A few pictures help a lot in this case.



The pictures are for $n = 12$ and $k = 2, 3$.

It looks like there 2-cycles and possibly fixed points, nothing else.

Geometry to the Rescue

Remember that the motions in D_n are either rotations or reflections with respect to a properly chosen axis, nothing else can happen.

As a consequence, for even n , there are either $n/2$ 2-cycles or $n/2 - 1$ 2-cycles and two fixed points: depending on whether the axis of the reflection passes through vertices or the center of the sides of the n -gon.

For odd n , there are always $(n - 1)/2$ 2-cycles plus one fixed point: the reflection axis always has to pass through a vertex and the center of one side.

Exercise

Draw pictures to confirm these assertions.

Putting it All Together

For simplicity, assume n is odd. Then the number of k -ary necklaces of length n is

$$\frac{1}{2n} \left(\sum_{p < n} |X_{\alpha^p}| + \sum_{p < n} |X_{\alpha^p \beta}| \right) =$$

$$\frac{1}{2n} \left(\sum_{p < n} k^{\gcd(n, p)} + \sum_{p < n} k^{(n+1)/2} \right) =$$

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

where φ is Euler's totient function: $\varphi(m) = |\mathbb{Z}_m^*|$.

The Even Case

For even n the counting result is very similar.

$$\frac{1}{2n} \sum_{d|n} \varphi(n/d) k^d + (k+1)k^{(n+1)/2}/4$$

Not too pretty, but no nice simple closed form is known. Enough, though, to compute some values. $k = 2, \dots, 6$ and $n = 1, \dots, 8$:

2	3	4	6	8	13	18	30
3	6	10	21	39	92	198	498
4	10	20	55	136	430	1300	4435
5	15	35	120	377	1505	5895	25395
6	21	56	231	888	4291	20646	107331

Back To Carbocycles

For our original problem, there are 13 possible molecules obtained from substituting some H atoms by OH groups.

Let's check.

(1, 1, 1, 1, 1, 1)
 (1, 1, 1, 1, 1, 2)
 (1, 1, 1, 1, 2, 2) (1, 1, 1, 2, 1, 2) (1, 1, 2, 1, 1, 2)
 (1, 1, 1, 2, 2, 2) (1, 1, 2, 1, 2, 2) (1, 2, 1, 2, 1, 2)
 (1, 1, 2, 2, 2, 2) (1, 2, 1, 2, 2, 2) (1, 2, 2, 1, 2, 2)
 (1, 2, 2, 2, 2, 2)
 (2, 2, 2, 2, 2, 2)

And Tic-Tac-Toe?

Can we answer the old Tic-Tac-Toe question at this point? We need to count the number of patterns of type (3, 3, 3).

So we have

- configuration space X consisting of all (3, 3, 3) placements,
- dihedral group D_4 acting on X .

More formally, $X \subseteq \{0, 1, 2\}^{3,3}$ is determined by the condition that the number of 0's, 1's and 2's in a matrix is exactly 3 each.

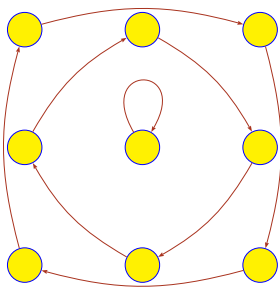
Also note that we need to deal with a subgroup of S_9 that is isomorphic to D_4 : the board has 9 squares.

We need to determine the cardinalities of the invariant sets. That requires a bit of fumbling.

Rotations

For rotations other than the identity all invariant sets are empty.

We only consider rotation by 90 degrees, the other cases are entirely similar.



To see that $X_\alpha = \emptyset$ note that since we only have 3 marks of each kind no 4-cycle can be invariant.

Note that this argument is a bit frail: if we were to look at different kinds of configurations we would need to start all over again – more on this later.

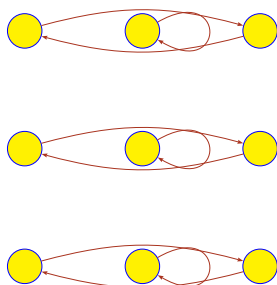
Exercise

Carry out the same argument for the invariant subset associated with α^2 , rotation by 180 degrees.

Also argue that $X_{\alpha^2} = \emptyset$ implies that $X_\alpha = \emptyset$.

Reflections

For reflections things are more interesting. We only consider reflection along the vertical axis, the other cases are entirely similar: the invariant set has size $6 \times 6 = 36$.



Final Reckoning

Hence the number of Tic-Tac-Toe configurations with 3 crosses and 3 noughts is

$$\frac{1680 + 36 + 36 + 36 + 36}{8} = 228$$

Not bad, but as already mentioned, this method becomes tedious if we ask about other types of boards. E.g., we have no idea how many patterns there are for 2 crosses and 2 noughts. It would be nice to have a global tool to handle all possible cases.

Exercise

Determine the largest invariant set (other than X_1 of course) for all possible configurations.

Summary

- Identification of similar objects can often be expressed as a group acting on a set.
- In this setting, one can count fixed points in two different ways, leading to Burnside's lemma.
- Burnside's lemma allows one to determine the number of patterns (equivalence classes) in many cases.
- Monoid actions have less structure but are also extremely important (e.g., DFAs can be considered as actions over the free monoid of all words over some alphabet).