

# CDM

## Actions and Counting

Klaus Sutner  
Carnegie Mellon University  
[www.cs.cmu.edu/~sutner](http://www.cs.cmu.edu/~sutner)

# Actions

## 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 1.** Let  $G$  be a group and  $X$  a set. A **left action of  $G$  on  $X$**  is a function  $\varphi$  such that

$$\varphi : G \times X \rightarrow X$$

$$\varphi(a * b, x) = \varphi(a, \varphi(b, x))$$

$$\varphi(1, x) = x$$

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

### Notation:

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

$$(a * b) \cdot x = a \cdot (b \cdot x)$$

$$1 \cdot x = x$$

This is much better notation, albeit slightly dangerous.

## A Fancy Way to Describe 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

$$\varphi(0, x) = f^0(x) = x$$

$$\varphi(k + l, x) = f^{k+l}(x) = f^k(f^l(x)) = \varphi(k, \varphi(l, x))$$

## 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)

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 next example, the permutations are written in one-line notation.

$$f = (2, 3, \dots, n, 1)$$

$$g = (2, 1, 3, \dots, n)$$

$$f \circ g = (1, 3, 4, \dots, n, 2)$$

$$(f \circ g) \cdot x = (x_1, x_3, \dots, x_n, x_2)$$

$$g \cdot x = (x_2, x_1, x_3, \dots, x_n)$$

$$f \cdot (g \cdot x) = (x_1, x_3, \dots, x_n, x_2)$$

**Exercise 1.** *Make sure you really understand the example, and then the general case. What would happen if we did composition the other way around?*

## And Right Actions

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

**Definition 2.** *Let  $G$  be a group and  $X$  a set. A **right action of  $G$  on  $X$**  is a function  $\varphi$  such that*

$$\varphi : X \times G \rightarrow X$$

$$\varphi(x, a * b) = \varphi(\varphi(x, a), b)$$

$$\varphi(x, 1) = x$$

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

To maintain sanity, we always  $a, b, c, \dots$  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.

$$\delta : Q \times \Sigma^* \rightarrow Q$$

$$\delta(p, \varepsilon) = p$$

$$\delta(p, xy) = \delta(\delta(p, x), y)$$

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.

## 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  $\Sigma^*$  (this time acting on the collection of all languages).

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

**Exercise 2.** *Check carefully that the last two examples really are right actions of the monoid  $\Sigma^*$ . Find better notation for the quotient operation.*

**Exercise 3.** *Find some natural examples for left monoid actions.*

## 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 1.** *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  $\varphi$  is a left action if, and only if,  $\psi$  is a right action.*

*Proof.* Suppose  $\varphi$  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  $\mathcal{G}^{\text{op}}$  is in fact a group.

Now any left action  $\varphi$  over  $\mathcal{G}$  translates into a right action  $\psi$  over  $\mathcal{G}^{\text{op}}$  by

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

**Exercise 4.** *Give a detailed proof of this claim.*

## Mathematics versus Implementation

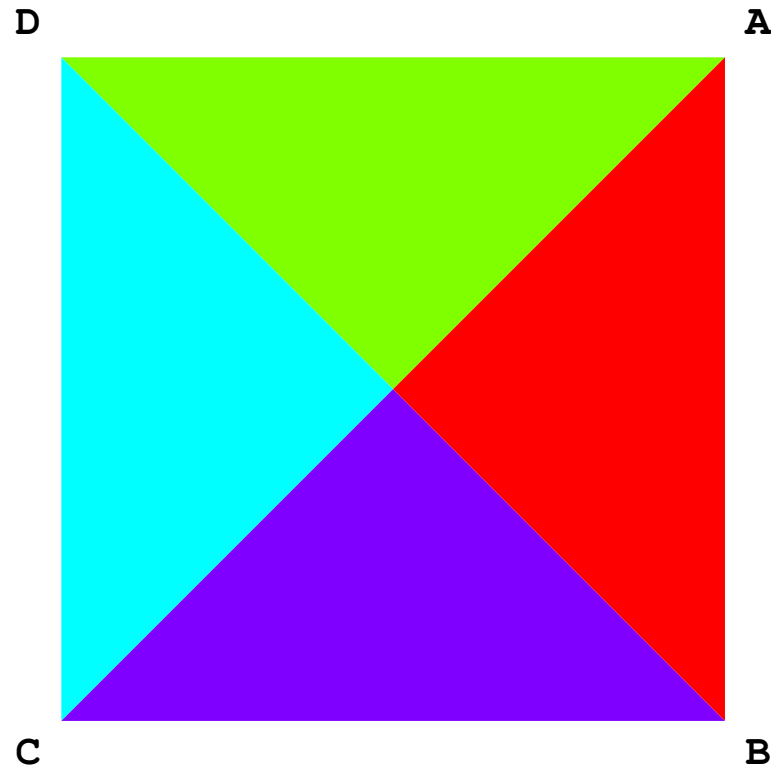
From a sufficiently abstract perspective, left and right actions are the same: it doesn't matter much if we replace each group element by its inverse or change the order of multiplication.

In fact, there are older texts that just speak about “a group acting on a set”. E.g., de Bruijn has an excellent introduction to Pólya counting 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

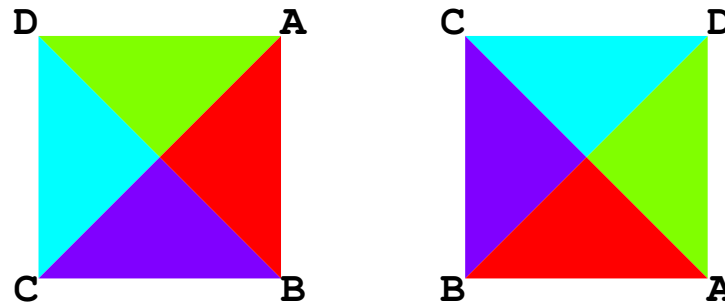
Consider the square with four vertices  $X = \{A, B, C, D\}$  in positions  $(\pm 1, \pm 1)$ .



## Dihedral Groups and Permutations

Every permutation  $\sigma$  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  $X$ .

**Definition 3.** *Let  $\varphi : G \times X \rightarrow X$  be a left action. The **orbit** of  $x \in X$  under  $G$  is defined to be*

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

*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 2.** *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.

# Burnside's Lemma

## Stabilizers and Invariants

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

**Definition 4.** Let  $\varphi : G \times X \rightarrow X$  be a group left action.

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

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

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

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

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 3.** *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.

□

## Orbit Sizes

**Proposition 4.** *The index  $[G : G_x]$  is the size of the orbit of  $x$ .*

*Proof.*

As already pointed out

$$a x = b x \iff b \in a G_x$$

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

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

□

## Enumeration

So if

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

where  $r = [G : G_x]$  then the orbit has the form

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

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  $g_i$ .

## 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 1.**

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

*Proof.*

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

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

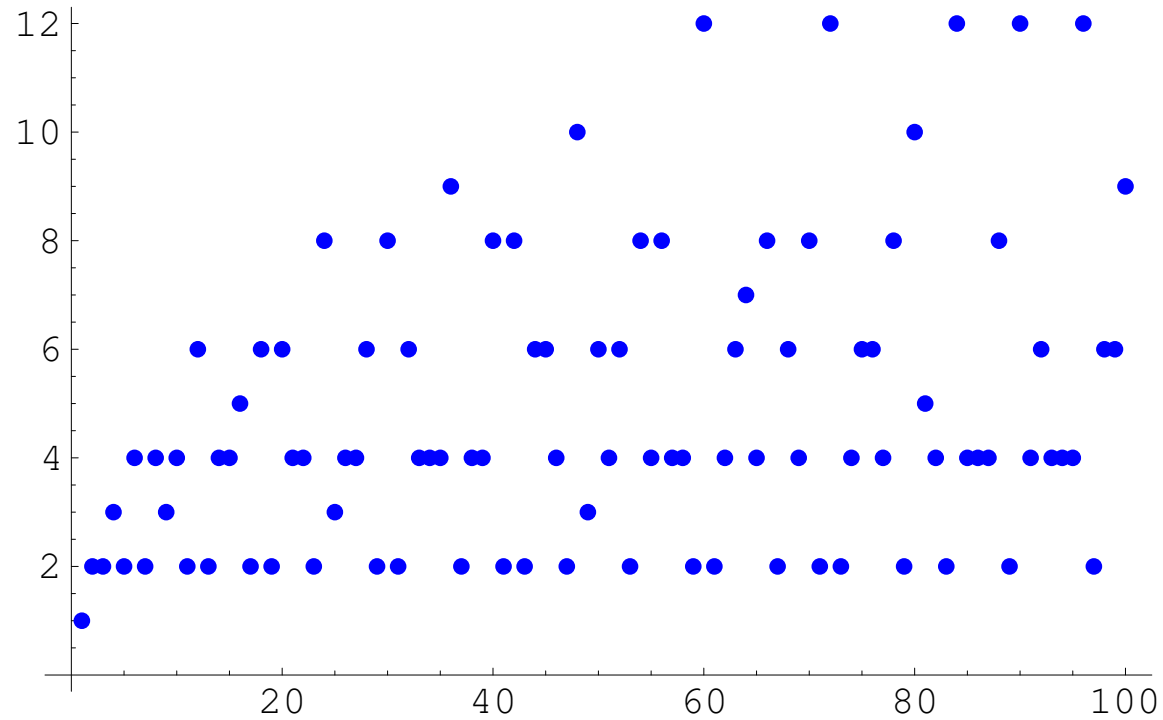
The rows are bitvectors for the invariant sets, and the columns are bitvectors for the stabilizers.

□

## Digression: Number of Divisors

For a positive integer  $n$  let  $d(n)$  be the number of divisors of  $n$ .

$d(n)$  is fairly complicated.



## Average Number of Divisors

How about the average

$$\widehat{d}(n) = 1/n \sum_{x \leq n} d(x).$$

Just as hopeless?

Let  $M$  be the  $n$  by  $n$  binary matrix with 1 in position  $(x, y)$  iff  $x$  divides  $y$ .

The number of 1's in column  $y$  is just  $d(y)$  and difficult.

But the number of 1's in row  $x$  is  $\lfloor n/x \rfloor$ .

So the total number of 1's is

$$\sum_{x \leq n} \text{row}(x) = \sum_{x \leq n} \lfloor n/x \rfloor \leq \sum_{x \leq n} n/x = n \cdot H_n.$$

and the error is at most  $n$ .

Hence  $\widehat{d}(n)$  is about  $\log n$ .

## Burnside's Lemma

**Theorem 1.** *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.*

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

□

## 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.

# 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

$$X = \mathbf{2}^n$$

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

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  $\{x, s x\}$  and has size 2.

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

## Application 1: Cellular Automata

A condensed version of the problem, details later: 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

$$X = \mathbf{2}^n$$

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

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

## The Invariants

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

$$X_1 = X$$

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

$$X_s = \emptyset$$

$$X_{rs} = \{ u(1-u)^R \mid u \in \mathbf{2}^{n/2} \}$$

Hence

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

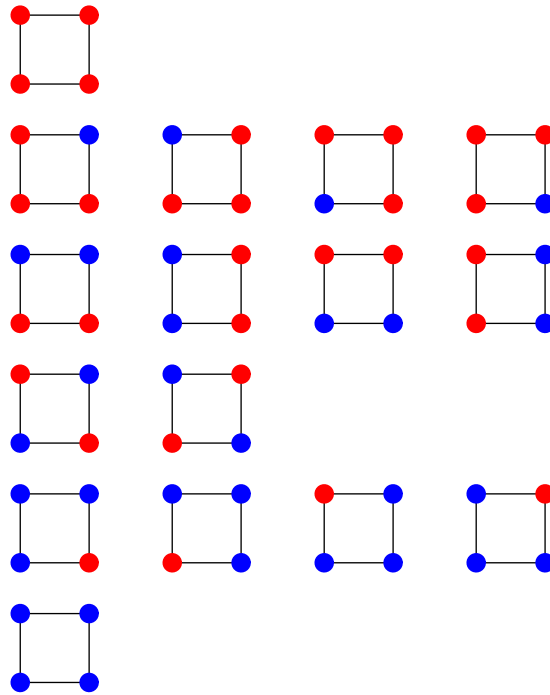
## Application 2: Coloring a Square

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  $D_4$ .

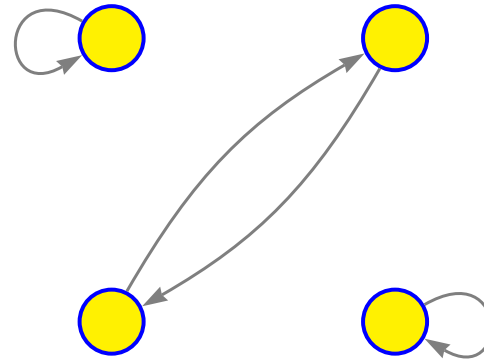
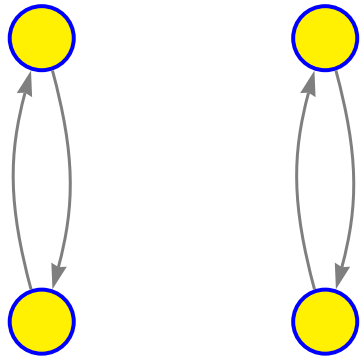
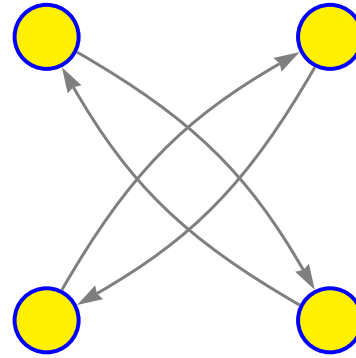
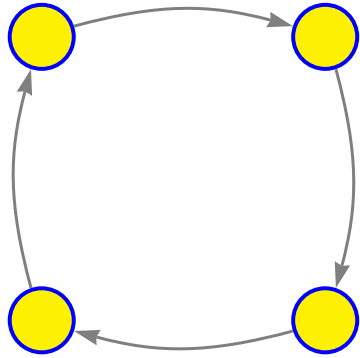
Hence

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

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

$$X_1 = X$$

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

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

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

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

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

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

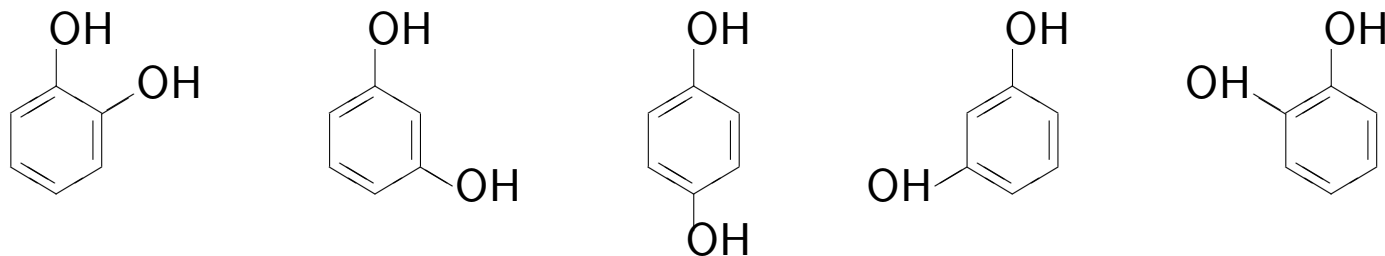
Hence

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

## 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

$$\begin{aligned} f_p : \mathbb{Z}_n &\rightarrow \mathbb{Z}_n \\ z &\mapsto z + p \pmod n \end{aligned}$$

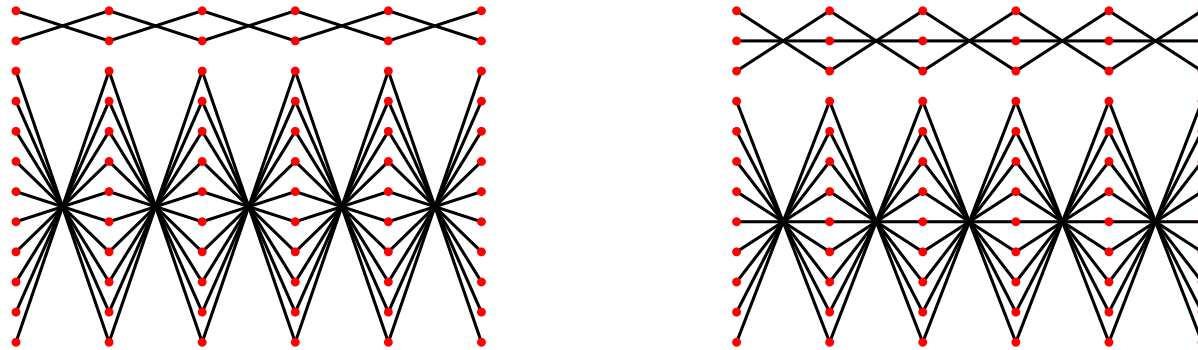
$f_p$  has  $\gcd(n, p)$  distinct orbits, each of length  $n / \gcd(n, p)$ .

$\alpha^p x = x$  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  $\alpha^p$ .

## 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 5.** *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

$$\begin{aligned} & \frac{1}{2n} \left( \sum_{p < n} |X_{\alpha^p}| + \sum_{p < n} |X_{\alpha^p \beta}| \right) = \\ & \frac{1}{2n} \left( \sum_{p < n} k^{\gcd(n,p)} + \sum_{p < n} k^{(n+1)/2} \right) = \\ & \frac{1}{2n} \sum_{d|n} \varphi(n/d) k^d + k^{(n+1)/2} / 2 \end{aligned}$$

where  $\varphi$  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.

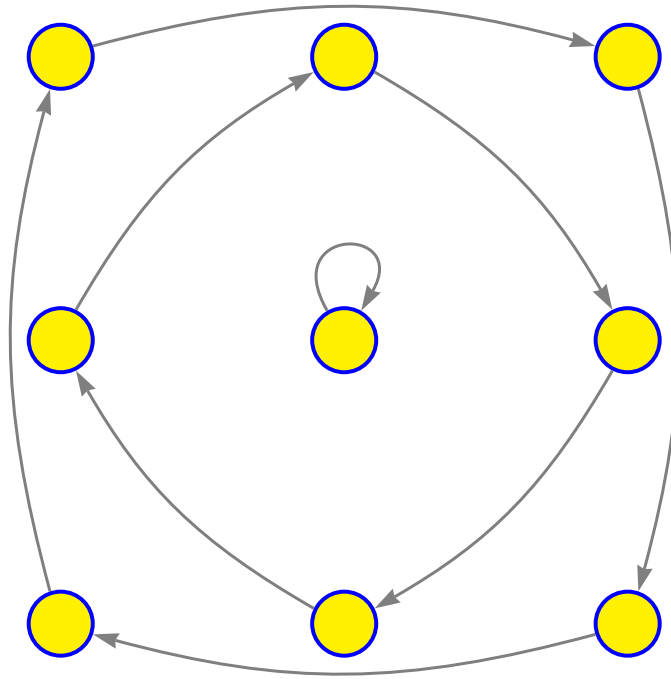
Also note that we need to deal with a subgroup of  $\mathbb{S}_9$  that is isomorphic to  $D_4$ : the board has 9 squares.

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.

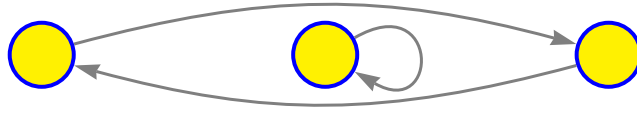
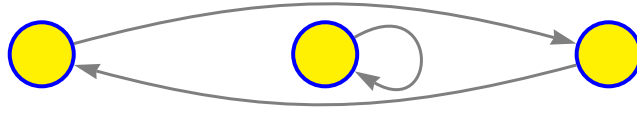
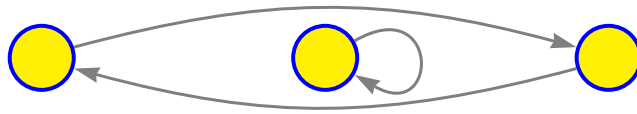
Note that this argument is a bit frail: if we were to look at different kinds of configurations we would need to start all over again – more on this later.

**Exercise 6.** *Carry out the same argument for the invariant subset associated with  $\alpha^2$ , 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 7.** *Determine the largest invariant set (other than  $X_1$  of course) for all possible configurations.*

# Pólya-Redfield

## Colors

Let's push the ideas from Burnside's Lemma a little bit further to make it easier to deal with Tic-Tac-Toe type questions. First, a slight abstraction.

Let

$$V = \{v_1, \dots, v_n\}$$

be a set of vertices (arbitrary objects), and

$$C = \{c_1, \dots, c_m\}$$

a *set of colors*. A *coloring* is a map  $V \rightarrow C$ .

We write  $X = V \rightarrow C$  for the set of all colorings.

## A Left Action

Let  $G$  be a subgroup of  $\mathbb{S}_n$  and define the natural left action of  $G$  on  $X$  by

$$\rho \cdot f = \rho \circ f$$

$$[n] \xrightarrow{\rho} [n] \xrightarrow{f} [k]$$

Thus,  $\rho \in G$  permutes the vertices, and then we apply the given coloring map.

What are the invariant elements  $f = \rho \cdot f$  under this action?

Clearly, if  $\rho$  is just a single cycle then  $f$  must be constant on the whole cycle.

In the general case, we consider the cycle decomposition.

## Generalize

Consider the cycle decomposition of  $\rho$ , including fixed points:

$$\rho = (v_{1,1}, \dots, v_{1,q_1}), (v_{2,1}, \dots, v_{2,q_2}), \dots, (v_{p,1}, \dots, v_{p,q_p})$$

Thus, the  $v_{i,j}$  are all distinct and  $\sum q_i = n$ .

We write

$$cc_i(\rho) = \text{number of cycles of length } i \text{ in } \rho$$

$$cn(\rho) = \sum cc_i(\rho)$$

for the total number of cycles in  $\rho$ .

So  $\sum_i i cc_i(\rho) = n$  and  $1 \leq cn(\rho) \leq n$ .

## Counting Invariants

**Lemma 2.** *The cardinality of  $X_\rho$  is  $m^{\text{cn}(\rho)}$ .*

*Proof.*

$f$  is in  $X_\rho$  iff  $f$  is constant on all the cycles of  $\rho$ .

But there are exactly  $m$  choices for the value of  $f$  on any one of the cycles.

□

Note, though, that this requires knowledge of the cycle number for each group element.

## Example: Dihedral Group

How about the cycle structure of all elements of  $D_n$ ?

For pure rotations  $\alpha^k$  the cycle structure is easy: there are  $\gcd(n, k)$  many cycles of length  $n / \gcd(n, k)$  each.

But, as we have seen, for reflections things are more complicated; we have to deal with 2-cycles and possibly fixed points.

For example, in an octagon there is a reflection (we write fixed points for clarity)

$$\rho = (1)(2, 8)(3, 7)(4, 6)(5)$$

## Counting Orbits

From Burnside's Lemma we immediately have the

**Corollary 1.** *The number of distinct orbits is  $N = \frac{1}{|G|} \sum_{\rho \in G} m^{\text{cn}(\rho)}$ .*

For example, when  $D_4$  is acting on the square we have

$\rho$	$\text{cn}(\rho)$	$\rho$	$\text{cn}(\rho)$
1	4	$\beta$	2
$\alpha$	1	$\alpha\beta$	3
$\alpha^2$	2	$\alpha^2\beta$	2
$\alpha^3$	1	$\alpha^3\beta$	3

Hence

$$N = \frac{1}{8}(m^4 + 2m^3 + 3m^2 + 2m).$$

## Pentagon

For  $D_5$  acting on the pentagon we get

$\rho$	$cc(\rho)$	$\rho$	$cc(\rho)$
1	5	$\beta$	3
$\alpha$	1	$\alpha\beta$	3
$\alpha^2$	1	$\alpha^2\beta$	3
$\alpha^3$	1	$\alpha^3\beta$	3
$\alpha^4$	1	$\alpha^3\beta$	3

Hence

$$N = \frac{1}{10}(m^5 + 5m^3 + 4m).$$

## Application: Boolean Functions

Let us identify two Boolean functions  $f$  and  $g$  if

$$f(x_1, x_2, \dots, x_k) = g(x_1 \oplus a_1, x_2 \oplus a_2, \dots, x_k \oplus a_k)$$

where the  $a_i$  are bits.

So  $a_i = 0$  means “leave the bit alone” and  $a_i = 1$  means “flip the bit”.

We want to count the Boolean functions modulo this equivalence.

As an example, there are 256 3-bit circuits. Flipping bits we could get the number down to  $256/8 = 32$  but that would require all variants to be distinct. So we should expect something like 50 (wild guess).

## Boolean Groups

As we will see shortly, the relevant group here is a Boolean group.

**Definition 5.** *A group is **Boolean** if every element other than the identity has order 2.*

Hence, in a Boolean group, we have

$$x + x = 0$$

for all  $x$  .

The additive notation is justified by the following exercise.

**Exercise 8.** *Show that every Boolean group is commutative.*

## Boolean Groups

**Example 1.** Let  $G = \langle \mathfrak{P}(A), \Delta, \emptyset \rangle$  where  $\Delta$  denotes symmetric difference:  $X\Delta Y = (X - Y) \cup (Y - X)$ . Then  $G$  is a Boolean group.

**Example 2.** Any finite Boolean group arises in the following way:

$$B_k = \langle \mathbf{2}^k, \oplus, \mathbf{0} \rangle$$

where  $\mathbf{2}^k$  means binary lists of length  $k$  and  $\oplus$  is point-wise exclusive or.

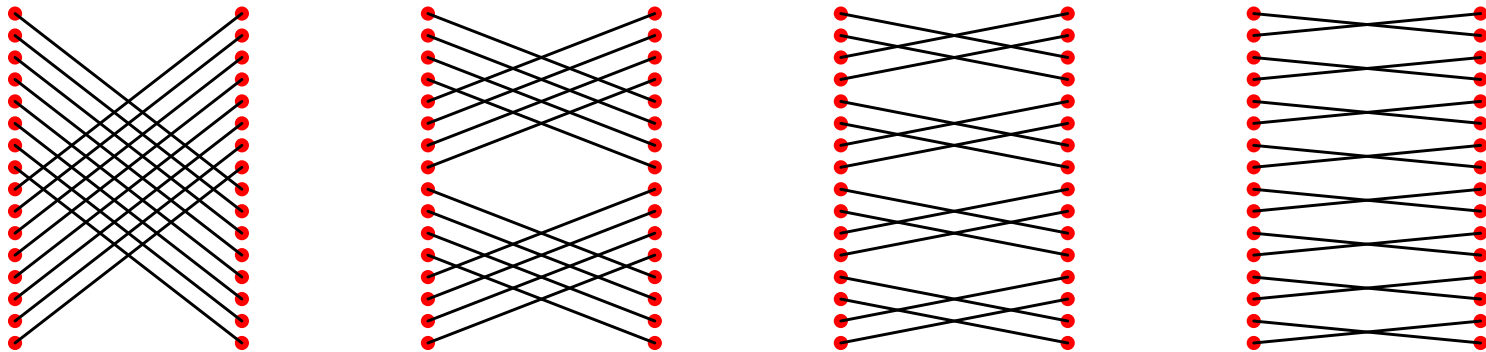
Note that  $B_k$  has  $k$  generators  $e_i = (0, \dots, 0, 1, 0, \dots, 0)$ .

## Embedding into a Permutation Group

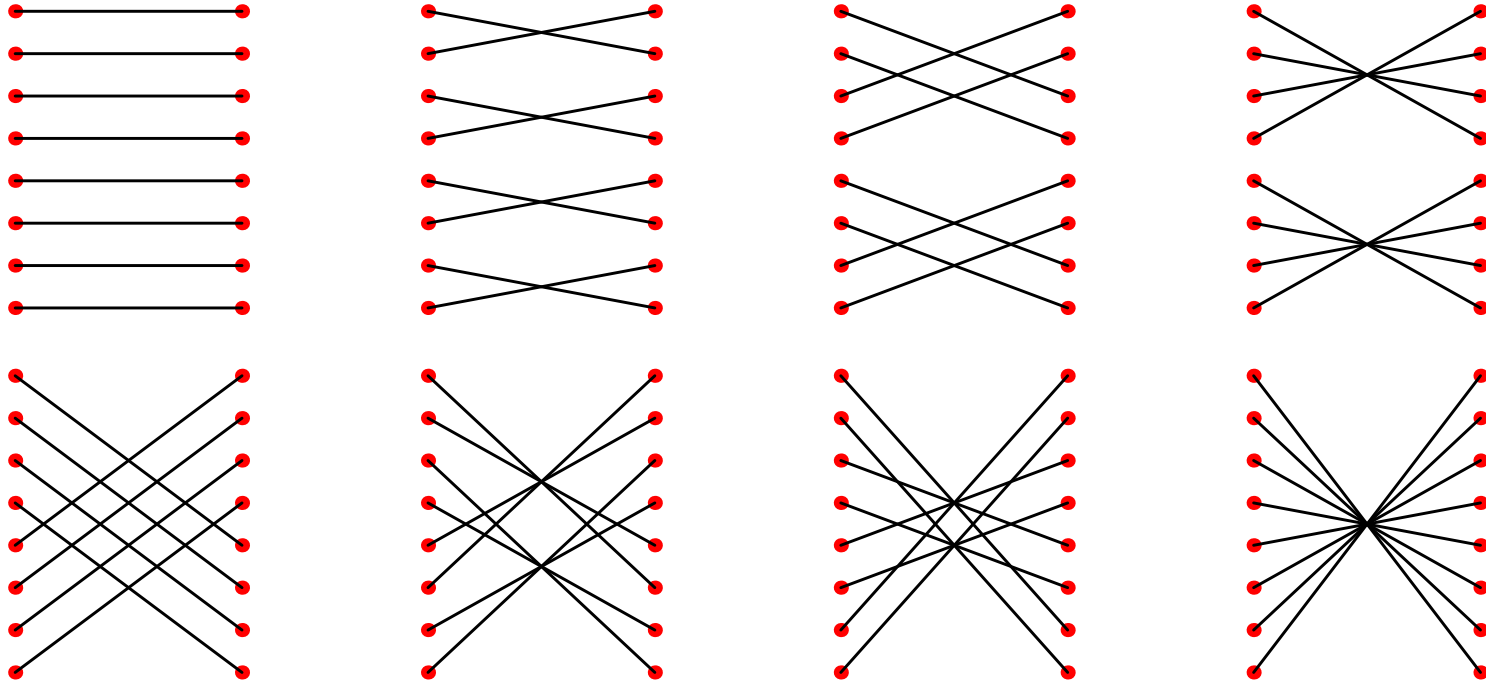
By Cayley's theorem we can identify  $B_k$  with a subgroup  $H$  of  $\mathbb{S}_{2^k}$ , the full permutation group on  $2^k$  points.

The permutation  $\hat{a}$  associated with  $a \in \mathbb{B}_k$  is the map  $x \mapsto x \oplus a$ .

For example, the pictures of the permutations  $\hat{e}_i$  for  $k = 4$  are shown below.



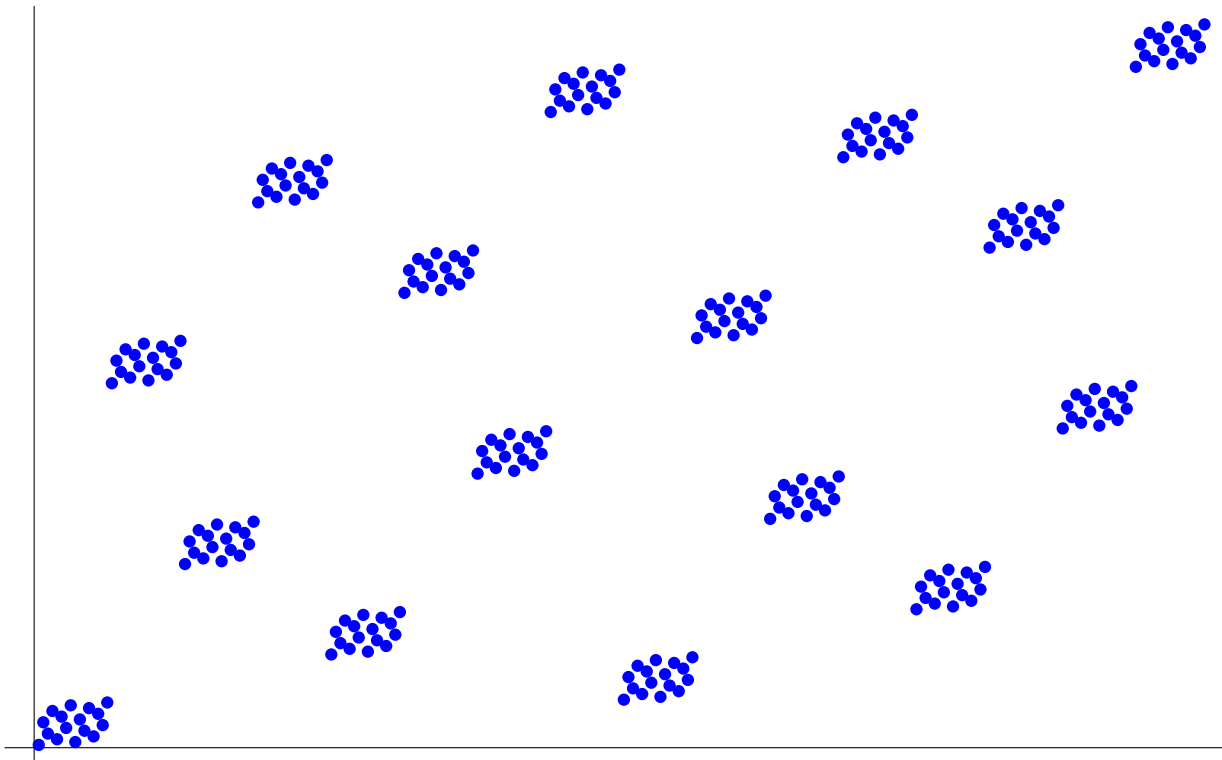
## All Permutations for $k = 3$



These are all the 8 permutations of the input values of a 3-bit circuit we can produce by flipping some of the bits.

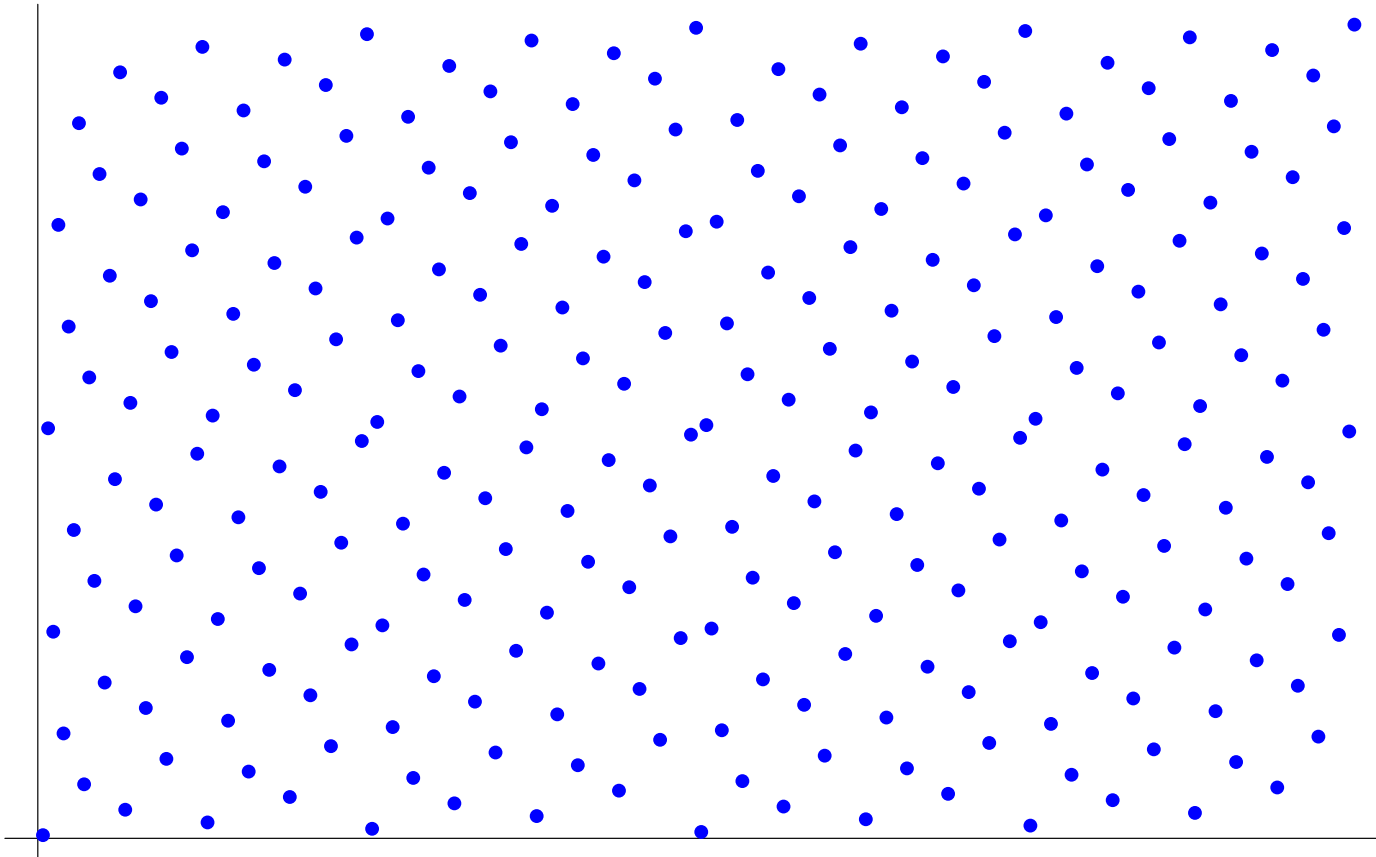
## Frivolous Picture

A 3-bit circuit can be described by a binary list of length 8 (representing the 8 outputs). Here is a picture of the permutation of these output vectors arising from flipping bits 2 and 3.



## Frivolous Picture 2

Same for flipping all bits. Somewhat surprising.



## Action

We can think of a Boolean function  $f : \mathbb{B}^k \rightarrow \mathbb{B}$  as a coloring of points  $V = \mathbb{B}^k$  by just two colors:  $C = \mathbb{B}$ .

Flipping some inputs in a circuit permutes the possible arguments to the corresponding Boolean function and the permutation  $\pi$  is an element in  $H$ .

But  $\pi$  is either the identity or has order 2. The number of fixed points is then

$$2^{2^k} \quad \text{and} \quad 2^{2^{k-1}}$$

To see this note that for a fixed point the coloring (value of Boolean function) must be the same on each 2-cycle of  $\pi$ .

This constraint cuts the degrees of freedom in half.

## Adding Up

Hence the total number of orbits is

$$1/2^k \left( 2^{2^k} + (2^k - 1)2^{2^{k-1}} \right) = 2^{2^{k-1}-k} \left( 2^{2^{k-1}} + 2^k - 1 \right)$$

Values for  $k = 1, \dots, 5$ :

3, 7, 46, 4336, 134281216

Thus, flipping input bits reduces the number of ternary Boolean functions from 256 to 46.

## Exercises

The last result deal with flipping input bits. Of course, there are several other natural operations we could modify a given circuit to obtain others – and combinations thereof.

- Flip output bit.
- Rotate inputs.
- Reverse inputs.
- Permute inputs.

**Exercise 9.** *Count the number of distinct circuits for some of these operations and combinations thereof.*

## Where Are We?

So far, we have a good tool to count the total number of orbits.

We even get a general formula that depends only the group  $G$  and applies to different kinds of configurations:  $m$  is just the number of colors, and the orbit count is a polynomial in  $m$ .

But we do not yet know how to determine the size of a specific orbit along the lines of the Tic-Tac-Toe problem (without a tedious special computation, that is).

To ameliorate this problem we will need to take a closer look at the number of cycles of each possible length:

$$cc_1(\rho), cc_2(\rho), \dots, cc_n(\rho).$$

## A Polynomial Trick

Define a monomial in  $n$  variables for each  $\rho \in G$ , *cycle index monomial* for  $\rho$ , by

$$Z_\rho(x_1, \dots, x_n) = x_1^{\text{cc}_1(\rho)} x_2^{\text{cc}_2(\rho)} \dots x_n^{\text{cc}_n(\rho)}$$

and the *cycle index polynomial* to be the sum of all these:

$$Z_G(\vec{x}) = \frac{1}{|G|} \sum_{\rho \in G} Z_\rho(\vec{x})$$

**Corollary 2.** *The number of distinct orbits is  $N = Z_G(m, m, \dots, m)$ .*

## Quoi?

Why should it be any better to write

$$x_1^{\text{cc}_1(\rho)} x_2^{\text{cc}_2(\rho)} \dots x_n^{\text{cc}_n(\rho)}$$

rather than the more pedestrian

$$\text{cc}_1(\rho), \text{cc}_2(\rho), \dots, \text{cc}_n(\rho)$$

The same information is conveyed both ways, we can easily translate back and forth.

True, but polynomials come equipped with a number of operations: addition, multiplication, substitution of integers, substitution of other polynomials.

As we will see, that's just what the doctor ordered.

## Example: $D_4$

For example, for  $G = D_4$  we have

$$Z_G(\vec{x}) = \frac{1}{8}(x_1^4 + 2x_1^2x_2 + 3x_2^2 + 2x_4).$$

Hence

$$N = Z_G(\vec{m}) = \frac{1}{8}(m^4 + 2m^3 + 3m^2 + 2m).$$

Addition, multiplication and substitution  $x_i \mapsto m$  all conspire to produce the right value.

Not bad, but we still can't quite handle the Tic-Tac-Toe problem in style.

## Weights

For  $f \in X$  let

$$dc_i(f) = \text{number of vertices } v \text{ such that } f(v) = c_i$$

Note that  $\sum_i dc_i(f) = n$ .

The weight of a configuration can be represented by a vector

$$(dc_1(f), dc_2(f), \dots, dc_m(f)) \in \mathbb{N}^m$$

and provides full information about the number of vertices of each color.

Weights are an invariant in the sense that all elements in an orbit of  $G$  have the same weight: the permutations can move the colors around, but they cannot change the color counts. So we can talk about the weight of an orbit.

## Formal Weights

Again, it turns out to be more convenient to define the *weight* of  $f$  by the monomial

$$W(f) = c_1^{\text{dc}_1(f)} c_2^{\text{dc}_2(f)} \dots c_m^{\text{dc}_m(f)}$$

Note that  $W(f)$  is a “purely formal expression”, the  $c_i$  are just colors.

More precisely,  $W(f)$  is an element of the polynomial ring  $\mathbb{Z}[c_1, \dots, c_m]$ .

Generating functions are your friend.

We will see shortly that this trick makes it easy to determine the number of orbits of a given weight.

## The Critical Example

Suppose we have  $n = 8$ ,  $\rho = (1)(2)(3)(4)(56)(78)$  where we have written the 1-cycles explicitly.

$$\text{Then } Z_\rho(\vec{x}) = x_1^4 x_2^2.$$

Which configurations over three colors  $r, g, b$  are invariant under  $\rho$ ?

All the cycles must be mono-chromatic, so the 2-cycles use up two occurrences of a color. Write  $+$  for logical or, and  $\cdot$  for choice. Then we have the possibilities

$$(r + g + b)(r + g + b)(r + g + b)(r + g + b)(r^2 + g^2 + b^2)(r^2 + g^2 + b^2) = (r + g + b)^4 (r^2 + g^2 + b^2)^2$$

**Major Insight:** this is just  $Z_\rho(r + g + b, r^2 + g^2 + b^2)$ .

## Expand

By expanding this polynomial we get a huge expression.

First assume the variables  $r, g, b$  to be non-commuting. Then there are 729 terms.

If we allow commutation, and collect monomials we get 45 terms, each of degree 8.

$$\begin{aligned}
 & b^8 + 4b^7g + 8b^6g^2 + 12b^5g^3 + 14b^4g^4 + 12b^3g^5 + 8b^2g^6 + 4bg^7 + g^8 + 4b^7r + \\
 & \quad 12b^6gr + 20b^5g^2r + 28b^4g^3r + 28b^3g^4r + 20b^2g^5r + 12bg^6r + 4g^7r + \\
 & \quad 8b^6r^2 + 20b^5gr^2 + 32b^4g^2r^2 + 40b^3g^3r^2 + 32b^2g^4r^2 + 20bg^5r^2 + 8g^6r^2 + \\
 & \quad 12b^5r^3 + 28b^4gr^3 + 40b^3g^2r^3 + 40b^2g^3r^3 + 28bg^4r^3 + 12g^5r^3 + 14b^4r^4 + \\
 & \quad 28b^3gr^4 + 32b^2g^2r^4 + 28bg^3r^4 + 14g^4r^4 + 12b^3r^5 + 20b^2gr^5 + 20bg^2r^5 + \\
 & \quad 12g^3r^5 + 8b^2r^6 + 12bgr^6 + 8g^2r^6 + 4br^7 + 4gr^7 + r^8
 \end{aligned}$$

## Specific Weights

Consider the term  $32 r^2 g^2 b^4$ .

The 32 choices for an invariant configuration of weight  $r^2 g^2 b^4$  are

*rrggbbbb, rrbgggbb, rrbbbbgg, rgrgbbbb, rggrbbbb, rbrbggbb, rbrbbbogg, rbbrrggbb, rbbrrbbgg, grrgbbbb, grgrbbbb, ggrrbbbb, ggbbrrbb, ggbbrrrr, bgbrrrbb, bgbrrrr, gbbgrrbb, gbbgrrrr, brrbggbb, brrbbbogg, brbrggbb, brbrbbgg, bggbrrbb, bggbrrrr, bgbrrrbb, bgbrrrr, brrrggbb, brrrbbgg, bbgrrrbb, bbgrrrr, bbbrrgg, bbbrrgrr*

These are precisely the words  $x$  over alphabet  $\{r, g, b\}$  with Parikh vector  $(2, 2, 4)$  subject to the constraint  $x_5 = x_6$  and  $x_7 = x_8$ .

## Logic and Algebra

This all hinges on the fact that we can express logic by algebra: we really need to consider choices of colors on the cycles of a given permutation.

But we can translate this counting problem into polynomial arithmetic by introducing formal variables for the colors and then working in the polynomial ring  $\mathbb{Z}[r, g, b]$ . Generating functions are amazingly powerful.

Since we can perform the arithmetic operations easily this translation allows for easy computation – sort of, a computer algebra system might come in handy.

Here is a more general description of this method.

## Pattern Inventory

Let

$$N(a_1, a_2, \dots, a_m) = \text{number of orbits with weight } c_1^{a_1} c_2^{a_2} \dots c_m^{a_m}.$$

and define the *pattern inventory* of  $G$  on  $X$  to be

$$\sum_{\vec{a}} N(a_1, a_2, \dots, a_m) c_1^{a_1} c_2^{a_2} \dots c_m^{a_m}$$

This is a polynomial in variables  $c_1, \dots, c_m$  whose coefficients contain exactly the counting information we are after.

Nice, but utterly useless unless we can somehow compute the inventory (without resorting to brute force, of course).

## Finale Furioso

**Theorem 2.** *Pólya-Redfield*

Set  $y_i = c_1^i + c_2^i + \dots + c_m^i$ . Then the pattern inventory is  $Z_G(y_1, y_2, \dots, y_n)$ .

Likewise,  $Z_\rho(y_1, y_2, \dots, y_n)$  is the generating function for the number of patterns of the indicated weight invariant under  $\rho$ .

It may still seem rather difficult to compute the pattern inventory, but with a little computer algebra it is not too hard.

## CIP for $D_4$

First we compute all the 8 monomials.

$\rho$	$Z_\rho$	$\rho$	$Z_\rho$
1	$x_1^4$	$\beta$	$x_2^2$
$\alpha$	$x_4$	$\alpha\beta$	$x_1^2 x_2$
$\alpha^2$	$x_2^2$	$\alpha^2\beta$	$x_2^2$
$\alpha^3$	$x_4$	$\alpha^3\beta$	$x_1^2 x_2$

Note that this depends solely on the group  $G \subseteq \mathbb{S}_n$  acting on the colorings, not on the colors themselves.

Hence we have to compute this only once.

## Two Colors

Assume  $m = 2$ .

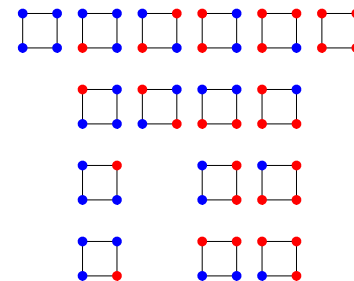
$$Z_{\alpha}(\vec{y}) = c_1^4 + c_2^4$$

$$Z_{\alpha^2}(\vec{y}) = c_1^4 + 2 c_1^2 c_2^2 + c_2^4$$

$$Z_{\alpha^3\beta}(\vec{y}) = c_1^4 + 2 c_1^3 c_2 + 2 c_1^2 c_2^2 + 2 c_1 c_2^3 + c_2^4$$

Lastly, here is the pattern inventory:

$$Z_G(\vec{y}) = c_1^4 + c_1^3 c_2 + 2 c_1^2 c_2^2 + c_1 c_2^3 + c_2^4$$

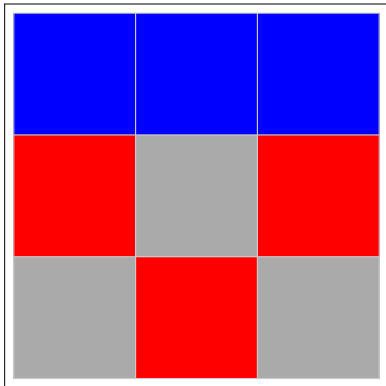


There are 6 orbits, two of weight  $c_1^2 c_2^2$ .

## Tic-Tac-Toe

For the Tic-Tac-Toe problem we have  $|V| = 9$  and  $|C| = 3$ .

As we have seen, the group  $G$  acting on the board is isomorphic to  $D_4$ , but is a subgroup of  $\mathfrak{S}_9$ .



Thus, the cycle decompositions for the permutations in  $G$  are quite different from the previous examples.

Too good to pass up as a homework problem.

## 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.
- Pólya's machinery pushes Burnside's lemma a bit further using polynomial arithmetic to obtain information about the size of specific orbits.
- 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).