Cellular Homology, simplified for now!

The ^ homology groups of a space: H_n

Choose your own adventure! Read these notes starting on this page for the idea of boundary, or start with page 3 (then come back to this point) if you want to see the definition of chain groups first. Here we start by motivating boundary maps via the fundamental group

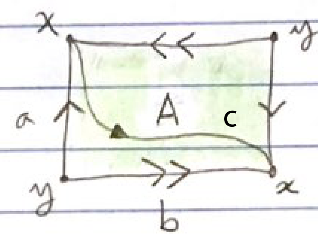
Recall the intuitive set concept of boundary:

$$\partial(D^2) = \partial(\text{disk}) = \bigcirc = S^1$$

$$\partial(S^1) = \emptyset \quad , \quad \partial(\cdot) = \emptyset$$

Note that the boundary of a boundary is empty. Also, the boundary of a line segment is two points.

Consider P^2: as a cell complex.



$$\chi(P^2) = 2 - 2 + 1 = 1$$

$$\pi_1(P^2) = \langle c \mid c = c^{-1} \rangle$$

→ Call the line segments a, b; corners x, y.

A is the interior 2-cell.

→ In the homology groups we get to use cells (disks) like D^0, D^1, D^2, D^3 as generators: they don't have to be loops. (like in π_1)

→ Also, we start by requiring that the group operation is commutative, like +, so abelian groups only. (But: the operation is not concatenation of loops anymore, as in π_1)

(Instead, the operation is purely formal---we just add two cells of the same dimension without any geometric meaning. See page 3)

→ So, we want a ∂ function on groups that does what it intuitively should. The key is to use an orientation of the disk. we pick counterclockwise.

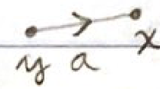
$$\partial(A) = \underbrace{a^{-1} b a^{-1} b}_{= e \text{ (identity) in } \pi_1} = \underbrace{-a + b - a + b}_{\text{In } H_1}$$

So, for a 2-cell, the boundary is the same thing as the relation in the fundamental group, in the format where it is solved for e. It's just written in additive notation. Again, we pick a starting point, which is arbitrary.

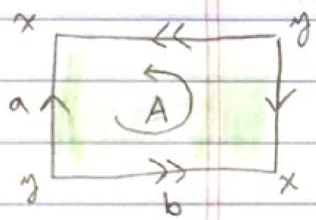
We need to define boundary for more than just 2-cells...

And we want to ensure that

$$\partial(\partial(A)) = 0$$

To do that, we define the boundary ∂ on line segments  with orientation

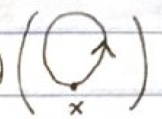
$$\partial(a) = x - y$$ (we choose to make the arrow point negative to positive)

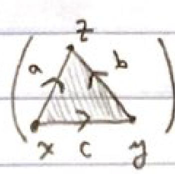


Note: we are already using D^0's (points) as generators of a group: we'll describe that soon!

Now, ∂ is defined to be homomorphism, in this case we can say ∂ is linear.

$$\begin{aligned}\partial(\partial(A)) &= \partial(-a+b-a+b)\\ &= -\partial(a)+\partial(b)-\partial(a)+\partial(b)\\ &= -(x-y)+x-y-(x-y)+x-y\\ &= 0\end{aligned}$$

Ex: $\partial(S^1) = \partial\Big($  $\Big) = x - x = 0$

Ex: $\partial(D^2) = \partial\Big($  $\Big) = c + b - a$

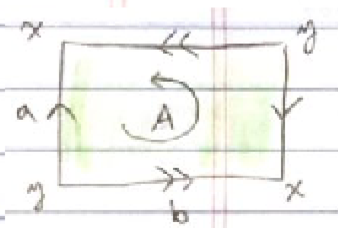
Note: the arrows here don't mean gluing, they are just arbitrary orientations. For the 2-cell we chose counterclockwise again, but it works either way.

Ex: $\partial(\partial(D^2)) = \partial(c) + \partial(b) - \partial(a) = y - x + z - y - (z - x) = 0$

Page 3: Alternate starting point.

Terminology

Given a space S, found as a cell complex made from points D^0, line segments D^1, 2-disks D^2, etc.., D^n (once glued into S, these n-disks are called n-cells) we say the $\boxed{\text{n-chains}}$ are linear combinations of the n-cells.



P^2

Ex: $a-b-a$	is	a 1-chain in P^2
$2a + 3b$	is	a 1-chain in P^2
b	is	a 1-chain in P^2
$x-y$	is	a 0-chain in P^2
$3x - 5y$	is	a 0-chain in P^2
A	is	a 2-chain in P^2
$-7A$	is	a 2-chain in P^2

For now we are using integer coefficients.

Let $\boxed{C_n(S)}$ be the free abelian group of n-chains. That means C_n is generated by the n-cells of S, and is commutative. "Free" means no relations other than $xx^{-1} = e$, which we write $x - x = 0$ since the operation is $+$.

That means that elements of C_n are finite linear combinations of the n-cells, with integer coefficients, and the operation is addition. -js

Another word for a free abelian group is a $\mathbb{Z}$-module (over the basis of cells.)

Now go back to page 1 if you skipped it.

There are two kinds of important chains:

Note: n is the dimension, so now there are "cycles" of any dimension, not just loops. Also, its the group-theory boundary, and 0+0 = 0. So although things that look like a loop, or a sphere, or a point are cycles, so are things like a pair of loops or a bunch of disconnected points or several spheres sharing sides like a bubble cluster.

n- $\boxed{\text{cycles}}$: these are n-chains that have 0 boundary

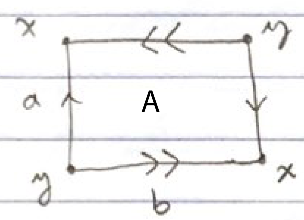
That is, finding the boundary gives 0.

n- $\boxed{\text{boundaries}}$: these are n-chains that are the boundary of a higher dimensional chain.

That is, we found them by taking a boundary.

Note: boundaries are special cycles, since $\partial(\partial(x)) = 0$. (We say $\partial^2 = 0$, the 0 map).

Ex:



- $b - a + b - a$ is a 1-cycle and a 1-boundary.
- $b - a$ is a 1-cycle, but not a 1-boundary

So is 7a-7b

1-cycles are made from loops!

$\partial(b-a) = x - y - (x - y) = 0$

- $b + 2a$ is neither
- x by itself is a 0-cycle but not a 0-boundary
- $x - y$ is both a 0-cycle and a 0-boundary

So is 5y-5x

- 0, the empty chain is both a cycle and a boundary ($C_{-1} = 0$)
- 3x + 2y is a 0-cycle but not a boundary.
- A is not a cycle, and not a boundary.

Big picture for a space S:
We add a subscript to ∂ to denote dim.

$$\cdots \longrightarrow C_3(S) \xrightarrow{\partial_3} C_2(S) \xrightarrow{\partial_2} C_1(S) \xrightarrow{\partial_1} C_0(S) \xrightarrow{\partial_0} 0$$

$\boxed{\text{chain complex.}}$ $\partial^2 = \partial_n \circ \partial_{n+1} = 0$

The $\boxed{\text{n-cycles}}$ are the $\boxed{\text{n-chains}}$ that get sent to 0 by ∂_n

So $\{\text{n-cycles}\} = \underline{\ker(\partial_n)}$. (or null-space of ∂_n)

The n-boundaries are the images of $(n+1)$-chains

So $\{\text{n-boundaries}\} = \underline{\mathrm{Im}(\partial_{n+1})}$. (or range ∂_{n+1})

Both of these are subgroups, (or sub-vector spaces if we use $\mathbb{R}$ for coefficients).
And $\boxed{\mathrm{Im}(\partial_{n+1}) \subseteq \ker(\partial_n)}$ since $\partial^2 = 0$.

Define $$\boxed{H_n(S) = \ker(\partial_n) \big/ \mathrm{Im}(\partial_{n+1})}$$

This is the quotient group.
It is seen by setting all the elements of $\mathrm{Im}(\partial_{n+1})$ equal to 0, in the larger group $\ker(\partial_n)$.

Note: we need a scheme for finding ∂_3 of 3-disks: simplices are useful!