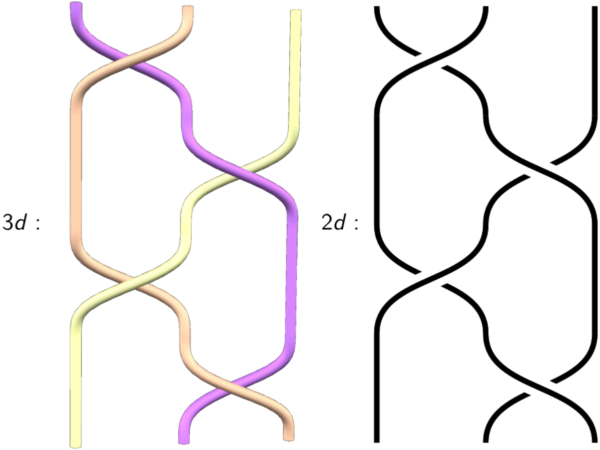
What is...the LKB representation?

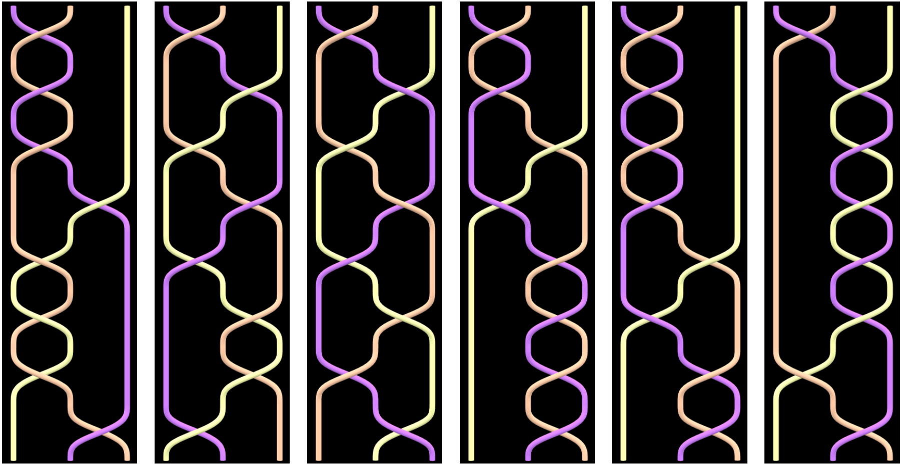
Or: Braids are linear

Braids



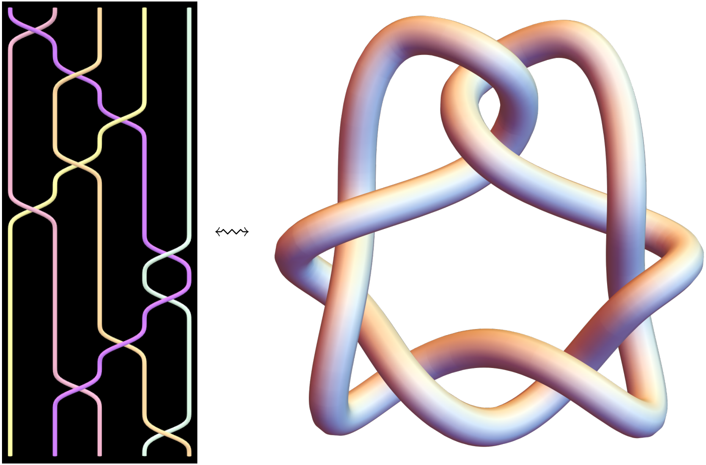
- Braids = strings in 3-space without turnbacks and with fixed bottom/top
- Two braids are the same if they are related by 3d isotopy
- Main question Can we determine whether braids are the same?

More braids



- The three braids above are all different but that is not obvious
- Braids can get arbitrary complicated
- Thus, it is hopeless to distinguish braids, right?

Alexander’s theorem



- Every braid gives a knot/link
- Distinguishing knots/links is very hard
- Thus, it is really hopeless to distinguish braids, right?

Enter, the theorem

Lawrence–Krammer–Bigelow (LKB) Braids are linear

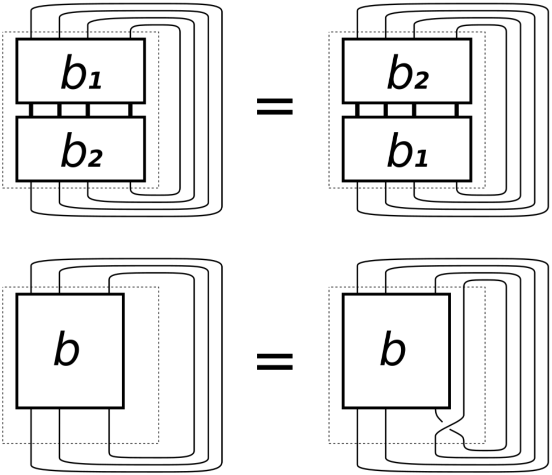
- I.e. there is a way to associate matrices $M(\beta)$ to braids β such that

$$\beta = \gamma \Leftrightarrow M(\beta) = M(\gamma)$$

```
sage: B = BraidGroup(3)
sage: b = B([1, 2, 1])
sage: b.LKB_matrix()
[             0 -x^4*y + x^3*y          -x^4*y]
[             0         -x^3*y               0]
[        -x^2*y  x^3*y - x^2*y               0]
sage: c = B([2, 1, 2])
sage: c.LKB_matrix()
[             0 -x^4*y + x^3*y          -x^4*y]
[             0         -x^3*y               0]
[        -x^2*y  x^3*y - x^2*y               0]
```

- This solves the braid recognition problem!
- Formally, the braid group has a faithful representation on a finite dimensional vector space

Knots are still hard



- There are two extra relations when going from braids to knots/links
- These extra relations ruin the recognition property

Thank you for your attention!

I hope that was of some help.