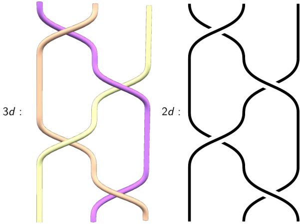
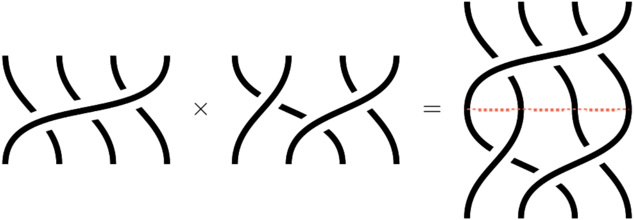
What is...braid group cryptography?

Or: Applications 2 (topology in cybersecurity)

Reminder: Braids and the braid group





- Braids = strings in $\mathbb{R}^3$ with endpoints fixed in two lines
- Stacking = groupoid structure on braids
- We get the braid group B_n with n = number of strands

Looking for "hard problems"

- **Conjugacy Decision Problem:** Given $u, w \in B_n$, determine whether they are conjugate, i.e., there exists $v \in B_n$ such that

$$w = v^{-1}uv$$

- **Conjugacy Search Problem:** Given conjugate elements $u, w \in B_n$, find $v \in B_n$ such that

$$w = v^{-1}uv$$

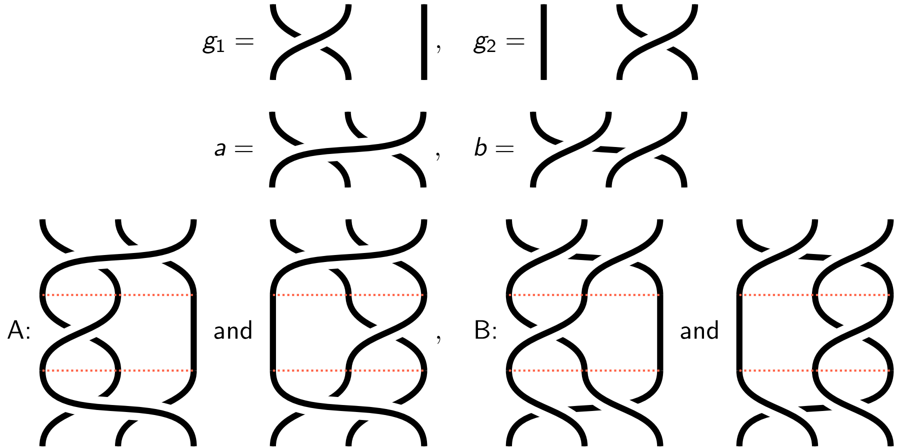
- **Multiple Simultaneous Conjugacy Search Problem:** Given m pairs of conjugate elements $(u_1, w_1), \ldots, (u_m, w_m) \in B_n$ which are all conjugated by the same element. Find $v \in B_n$ such that

$$w_i = v^{-1}u_iv, \quad \forall i \in \{1, \ldots, m\}$$

- **Decomposition Problem:** $u \notin G \leq B_n$. Find $x, y \in G$ such that $w = xuy$.

- For cryptography we want problems that are hard to solve
- **Example.** The conjugacy search problem "looks difficult"
- Idea Encode a public key exchange in B_n based on one of the above

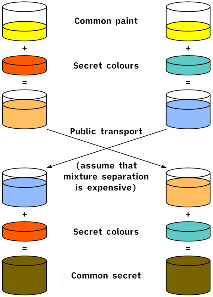
An example protocol (following Anshel–Anshel–Goldfeld)



- The common secret is $\langle g_1, \dots, g_m \rangle \leq B_n$, party A chooses $a \in \langle g_1, \dots, g_m \rangle$ and B chooses $b \in \langle g_1, \dots, g_m \rangle$
- Party A sends $a g_i a^{-1}$ and party B sends $b g_i b^{-1}$ for $i = 1, \dots, m$
- The common secret is $aba^{-1}b^{-1}$: A gets $ba^{-1}b^{-1} = bg_{i_1}b^{-1} \dots bg_{i_k}b^{-1}$ and ditto for B

For completeness: A formal statement

The AAG key-exchange protocol (previous slide) was proposed for B_{80} 80 strands!



The difficulty depends on the (multiple) conjugacy search problem in B_n

- There are variants based on other "hard problems"
- The braid group cryptosystems can be attacked (next slide)
- However, varies other "topological meaningful" groups can be used and are still not attacked

Attacking braids

- 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!

- The braid group have very efficient matrix representations *e.g.* LKB
- These can be used to attack the braid group cryptosystems (at least partially)
- Idea Solve the conjugacy search problem in matrices and lift the solutions to B_n

Thank you for your attention!

I hope that was of some help.