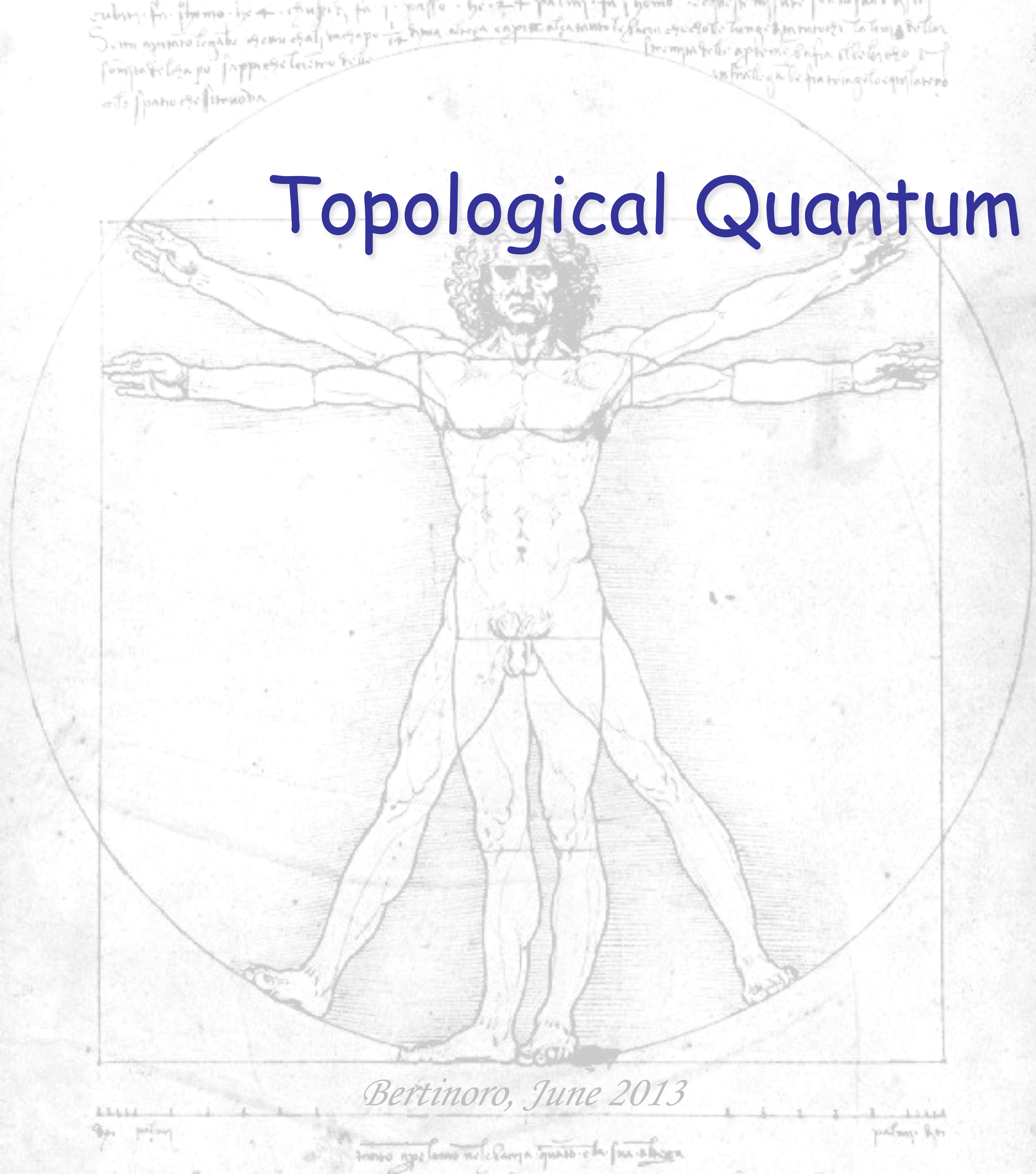
Topological Quantum Computation

Jiannis K. Pachos

Introduction



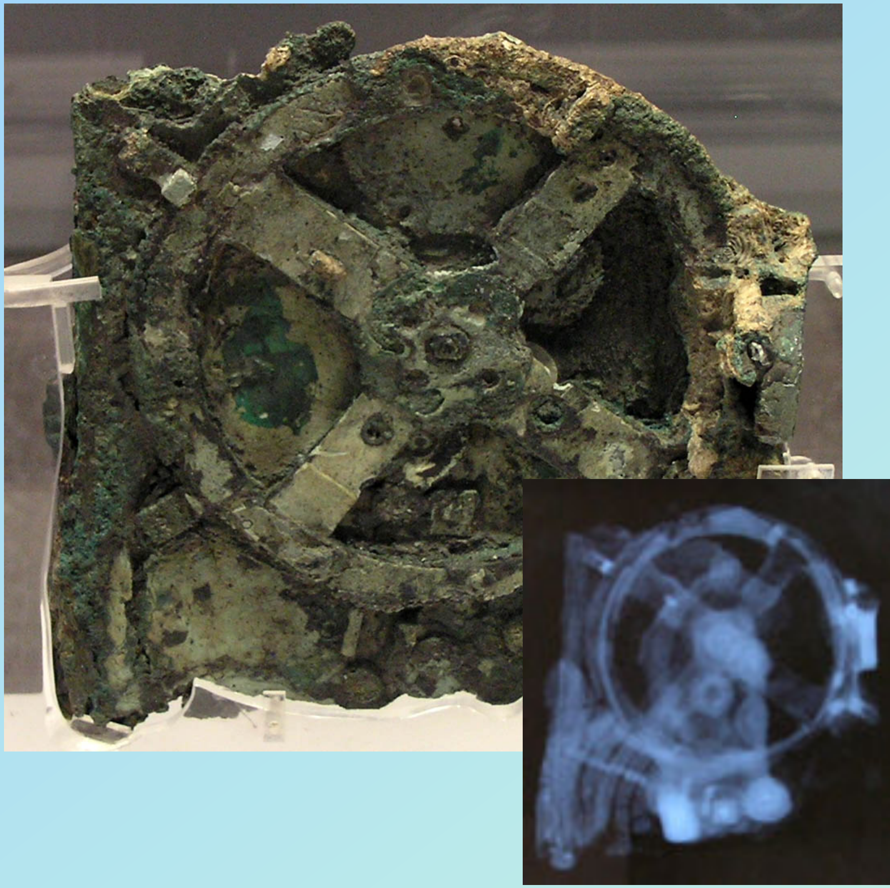
Bertinoro, June 2013

EPSRC
Engineering and Physical Sciences Research Council

UNIVERSITY OF LEEDS

Computers

Antikythera mechanism



Robotron Z 9001

Analogue computer

Digital computer: 0 & 1

Quantum computers

- **Quantum** binary states $|0\rangle, |1\rangle$ (spin,...)

$$|\Psi\rangle = a|0\rangle + b|1\rangle$$ (*quantum superposition*)

$$|a|^2 + |b|^2 = 1$$

If you **measure** $|\Psi\rangle$ then $P_0 = |a|^2, \;\; P_1 = |b|^2$

This is the quantum bit or "**qubit**"

- Many qubits (two):

$$|\Psi\rangle = |0\rangle \otimes |0\rangle$$ product state (classical)

$$|\Psi\rangle = a|0\rangle \otimes |0\rangle + b|1\rangle \otimes |1\rangle$$ entangled state

Quantum computers

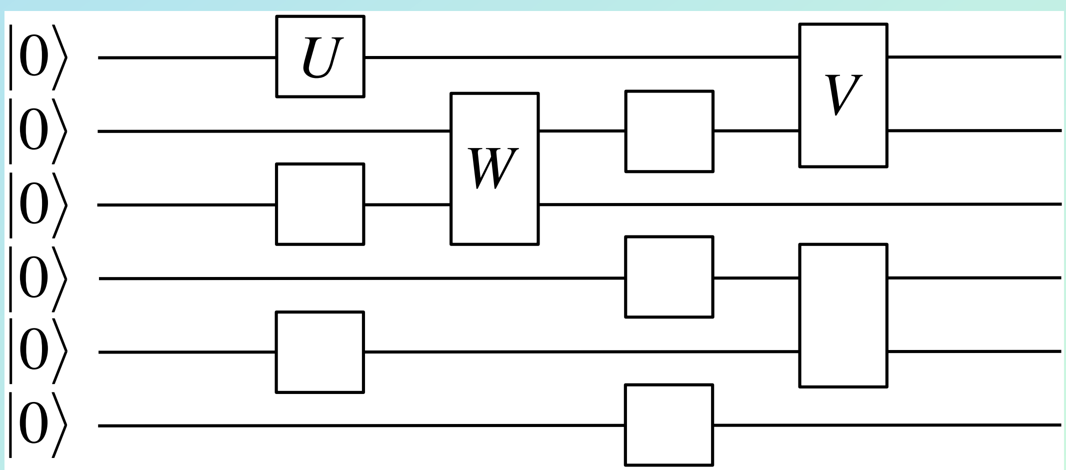
- Quantum binary gates-> **Unitary matrices**

$$U_{1q}|0\rangle = a|0\rangle + b|1\rangle$$

$$U_{2q}|0\rangle \otimes |0\rangle = a|0\rangle \otimes |0\rangle + b|1\rangle \otimes |1\rangle$$

- **Universality**: Any U_{1q} and a $U_{2q} \Rightarrow$ any q. algorithm.
- Quantum circuit model:

Input (product state) Initiate



Measure (product state) Output

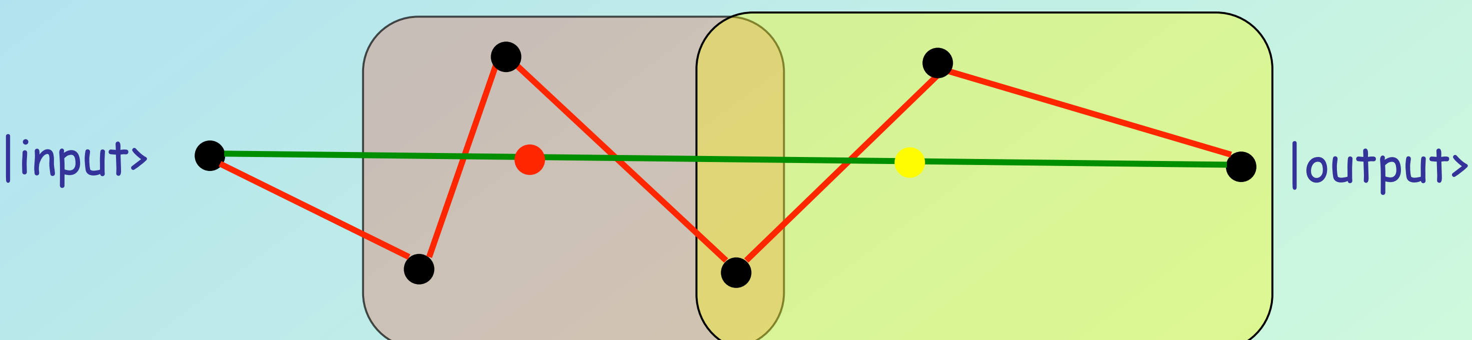
Quantum computers

- Quantum Algorithm:

Input
(product state)
Initiate

Measure
(product state)
Output

$$|\Psi_{\text{output}}\rangle = U_n \ldots U_2 U_1 |\Psi_{\text{input}}\rangle$$



Quantum computers: Why?

- **Computational complexity**

  Problems that can be solved in:

  -polynomial time (easy)

  -exponential time (hard)

  as a function of input size.

- Classical computers:

  P: polynomially easy to solve

  NP: polynomially easy to verify solution

- **BQP**: polynomially easy to solve with QC

Quantum computers: Why?

- **Factoring (Shor)**

  1807082088687404805951656164405905566278102516769401349170127021450056662540244048387341127590812303371781887966563182013214880557
  =
  39685999459597454290161126162883786067576449112810064832555157243
  ×
  45534498646735972188403686897274408864356301263205069600999044599

  quantum hackers *exponentially* better than classical hackers!

- **Searching** objects **(Grover)**: where is ❥?

  ¢®¶ⓘ⊙↓♫|ΠÃ≥⅙⃠⏎✜➚✍🕔Ψ”✎👓♈✪❥⅖ű

Quantum computers: Why?

- **Factoring** algorithm **(Shor)**:
  - Exponentially faster than known classical algorithm, but we do not know if there is a better classical one…
- **Searching** algorithm **(Grover)**:
  - Quadratic speed up (optimal), does not change complexity class…
- Still important enough: worth investigating…
- **Errors** during QC are **too catastrophic**.

Topological quantum computers: Why?

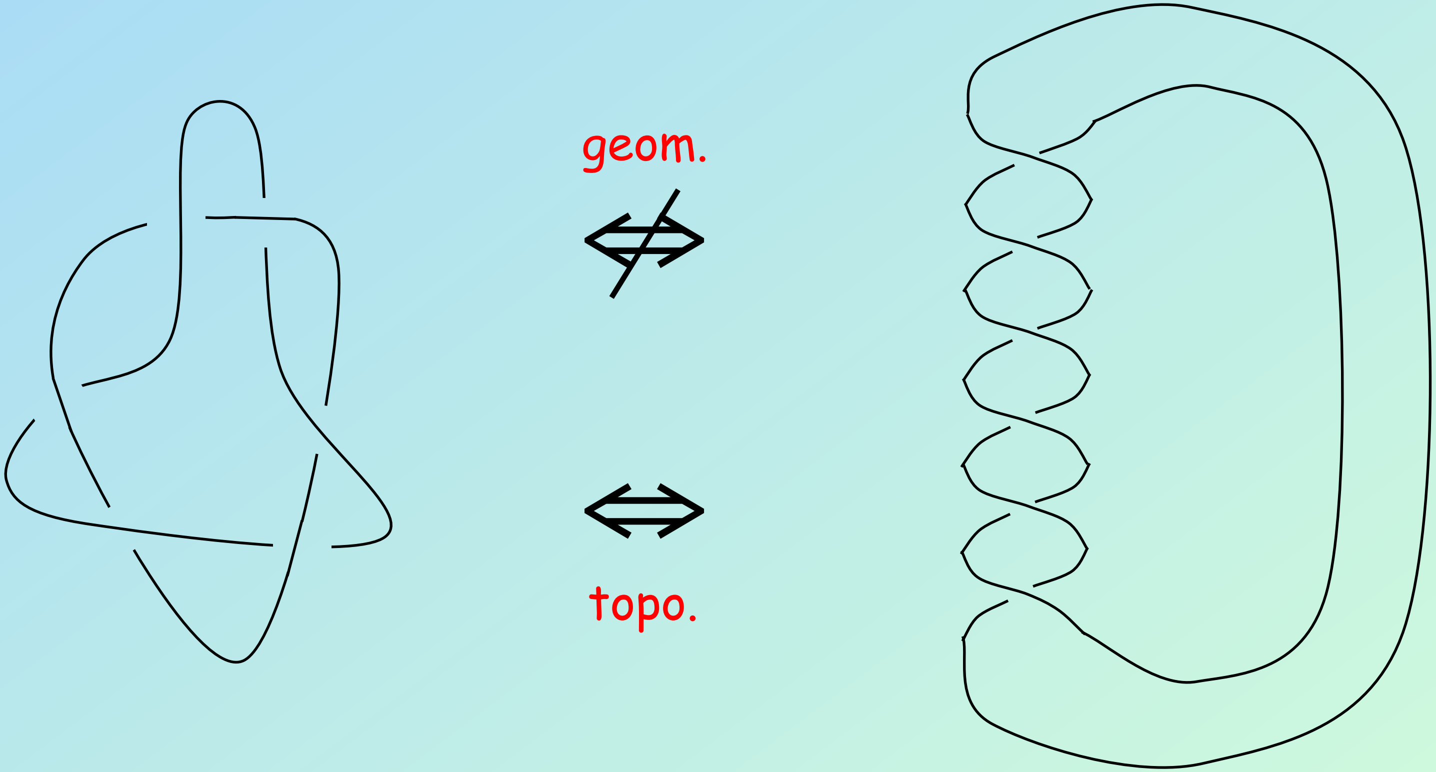



Topology promises to solve the problem of **errors** that inhibit the experimental **realisation of quantum computers**…

…and it is a lot of fun :-)

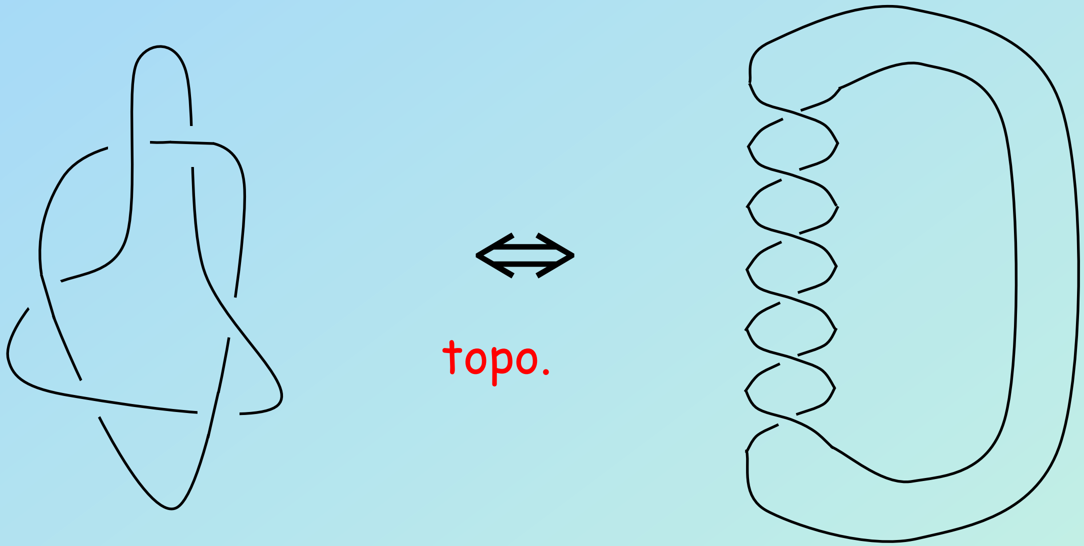
Geometry - Topology

- Geometry
  - Local properties of object
- Topology
  - Global properties of object



Topology of knots and links

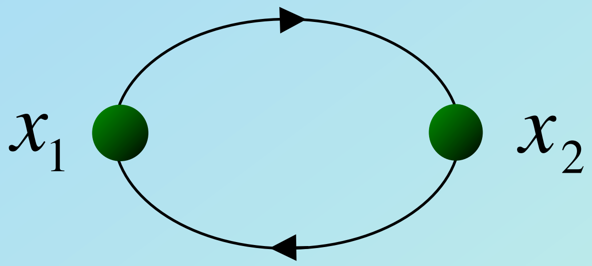
Are two knots equivalent?



- Algorithms exist from the '60s
- **Extremely** time consuming…
- Common problem (speech recognition, …)
- *Mathematically* ***Jones polynomials*** *can recognise if two knots are inequivalent.*

Particle statistics

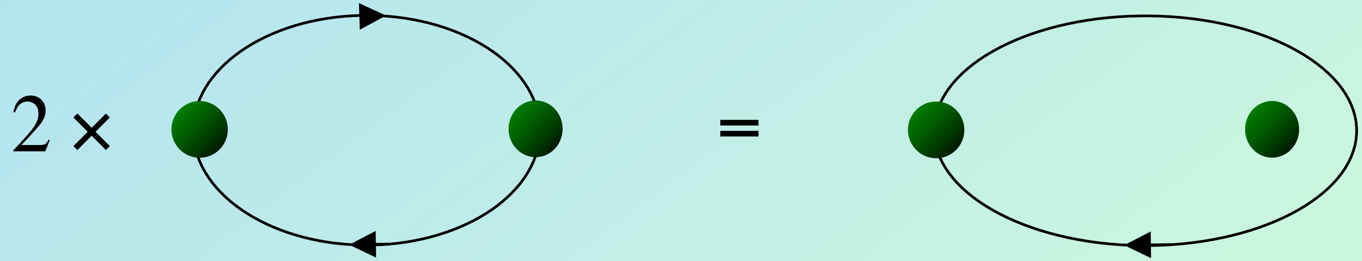
Exchange two **identical** particles:



Statistical symmetry:

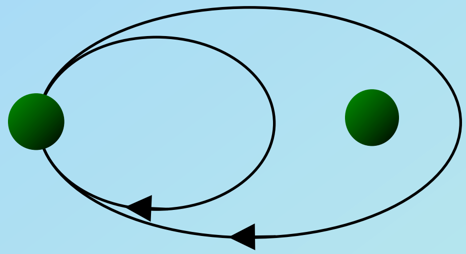
Physics stays the same, but $|\Psi\rangle$ could change!

$$|\Psi(x_1, x_2)\rangle = ???\,|\Psi(x_2, x_1)\rangle$$



Anyons and statistics

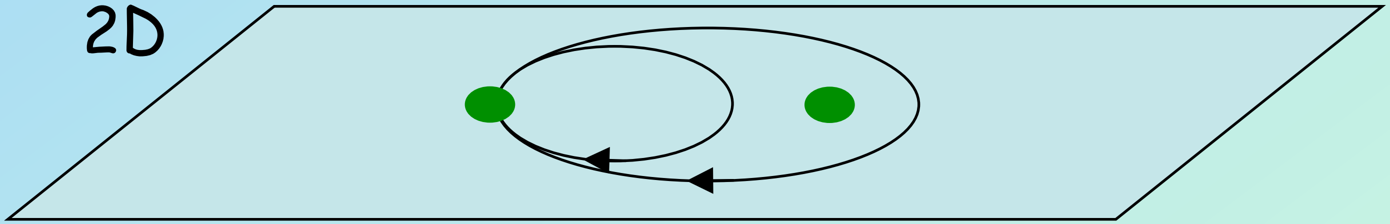
3D



Bosons $|\Psi\rangle \rightarrow |\Psi\rangle$

Fermions $|\Psi\rangle \rightarrow e^{i2\pi}|\Psi\rangle$

2D



$|\Psi\rangle \rightarrow e^{i2\phi}|\Psi\rangle$

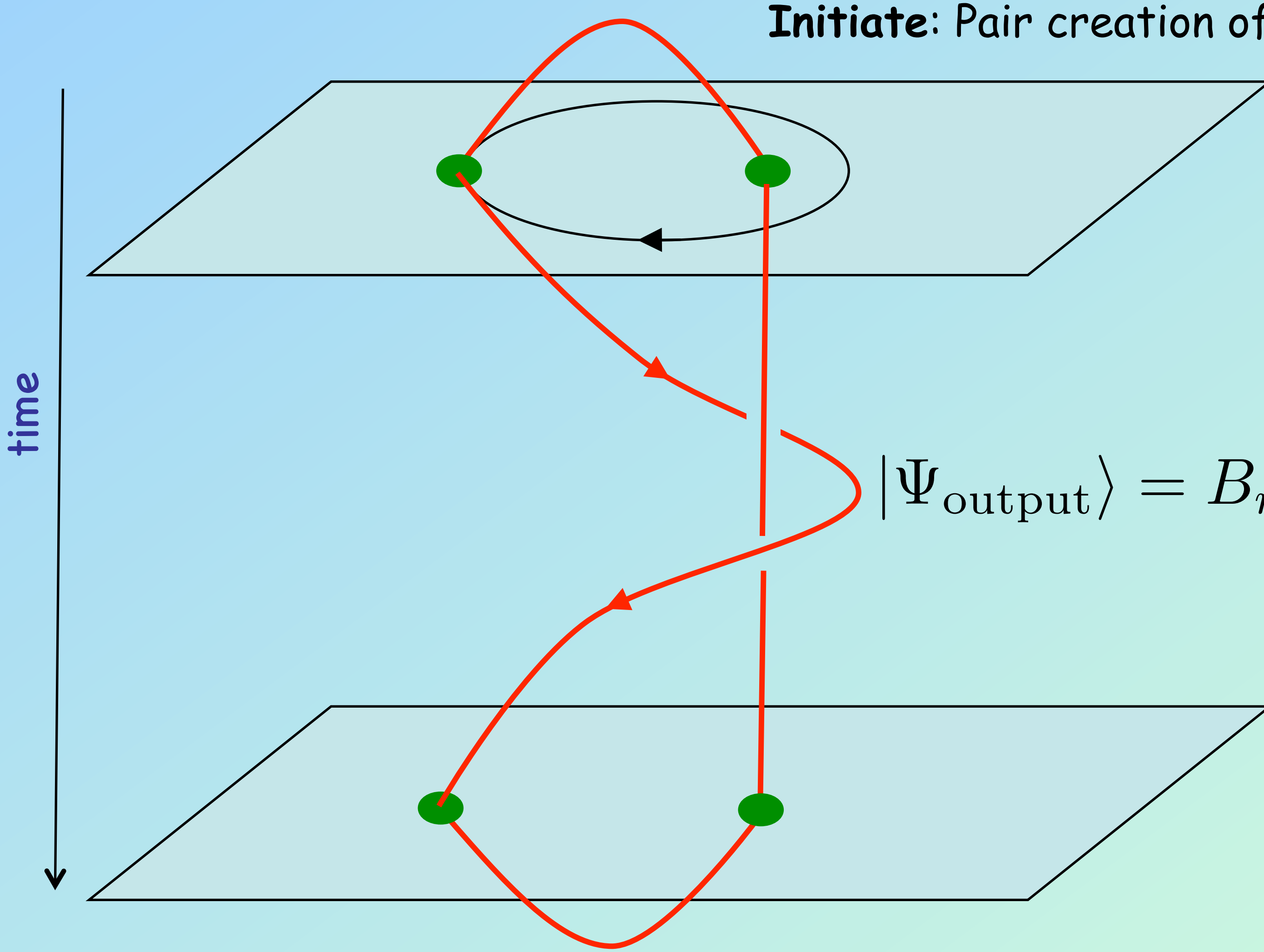
$|\Psi\rangle \rightarrow B|\Psi\rangle$

Anyons

Like a quantum gate

Anyons, statistics and knots

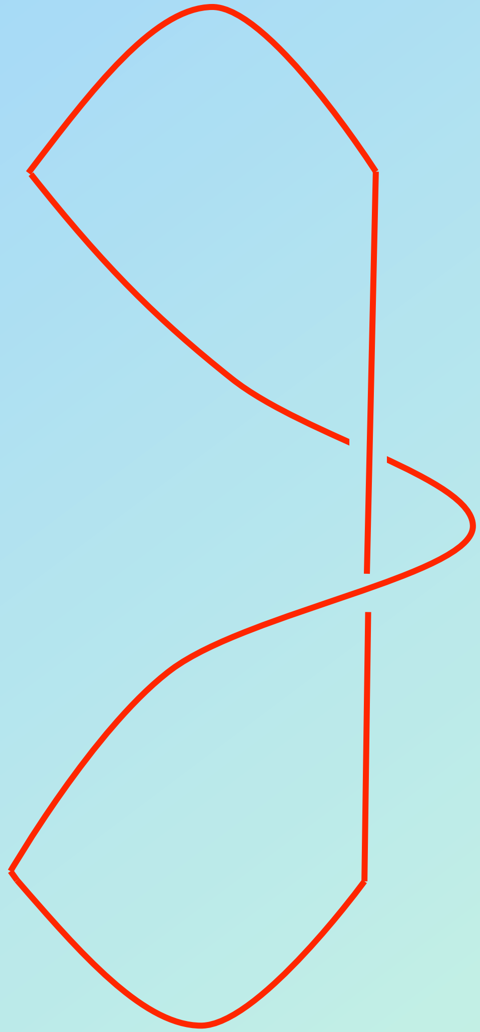
Initiate: Pair creation of anyons



$$|\Psi_{\text{output}}\rangle = B_n \ldots B_2 B_1 |\Psi_{\text{input}}\rangle$$

Measure: do they fuse to the vacuum?

Anyons, statistics and knots

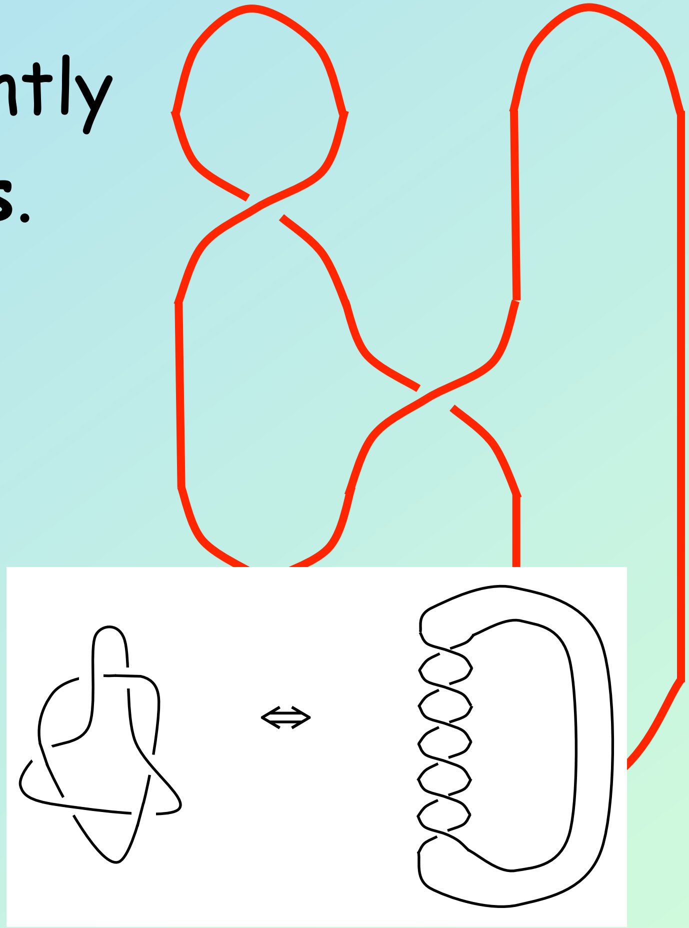


$$|\Psi_{\mathrm{output}}\rangle = B_n ... B_2 B_1 |\Psi_{\mathrm{input}}\rangle$$

Anyons and knots

Assume I can generate anyons in the laboratory.

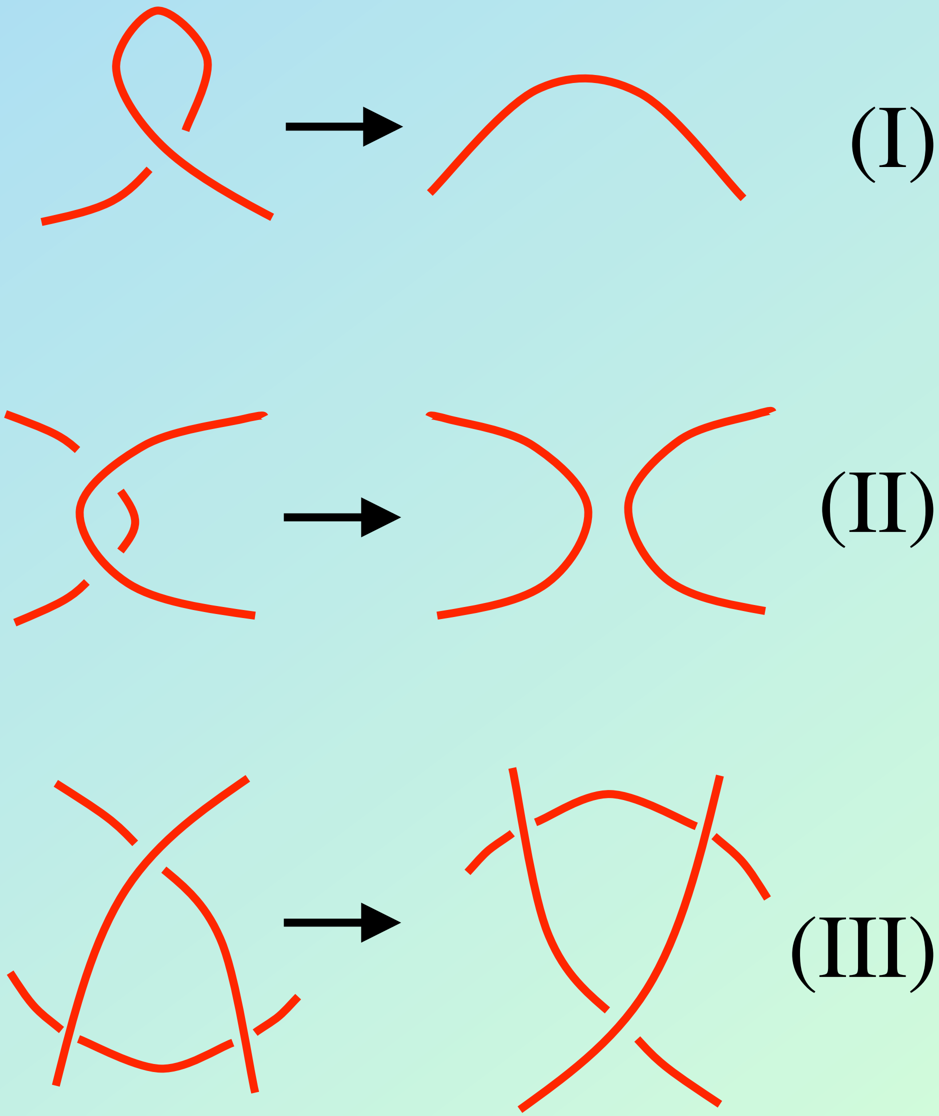
- The state of anyons is efficiently described by their **world lines**.
- Creation, braiding, fusion.
- The final **quantum state** of anyons is **invariant** under **continuous** deformations of strands.



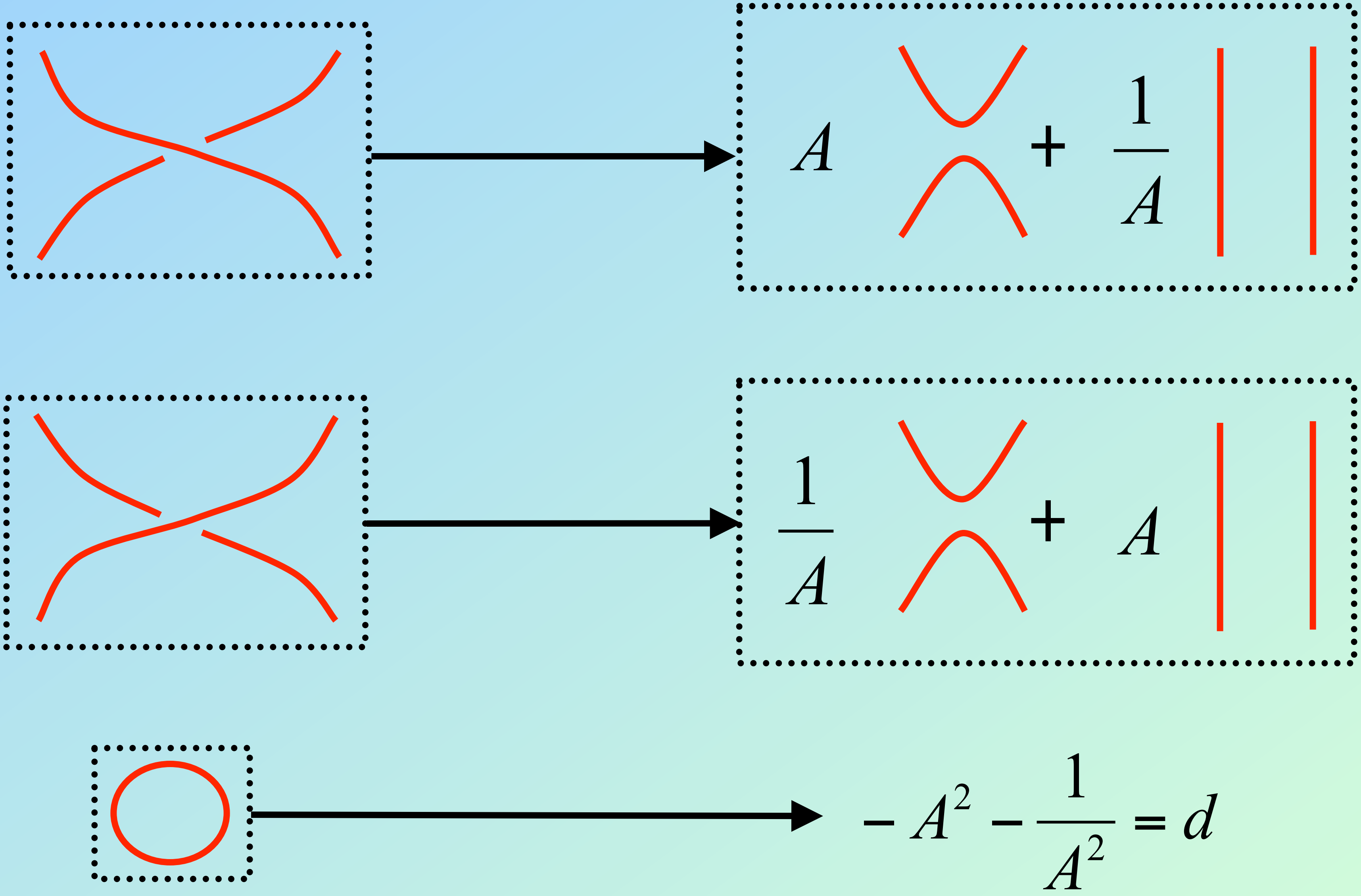
The Reidemeister moves

Theorem:

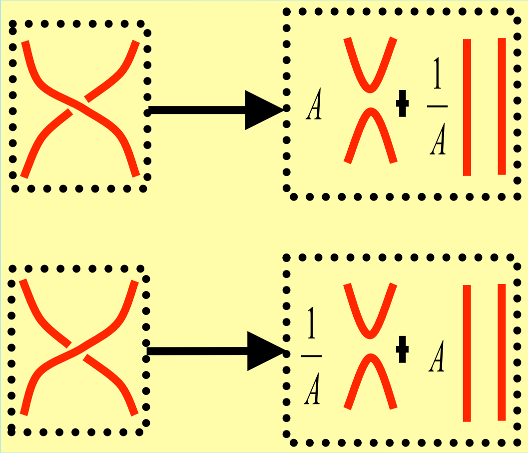
Two knots can be **deformed continuously** one into the other iff one knot can be transformed into the other by **local moves**:

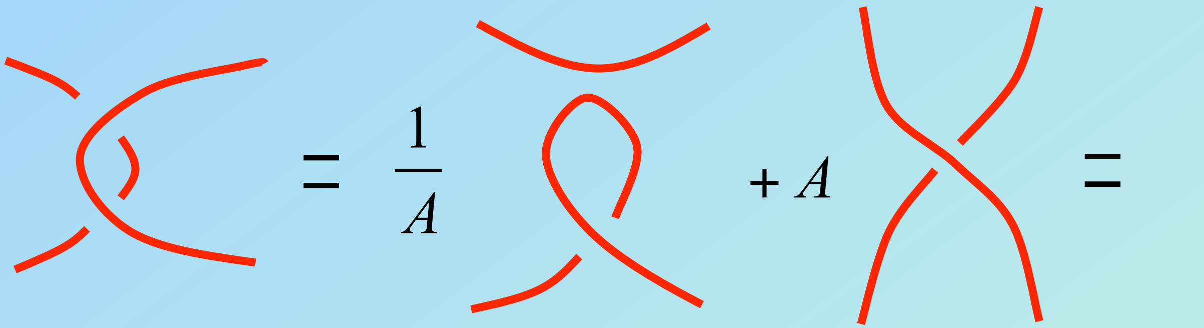


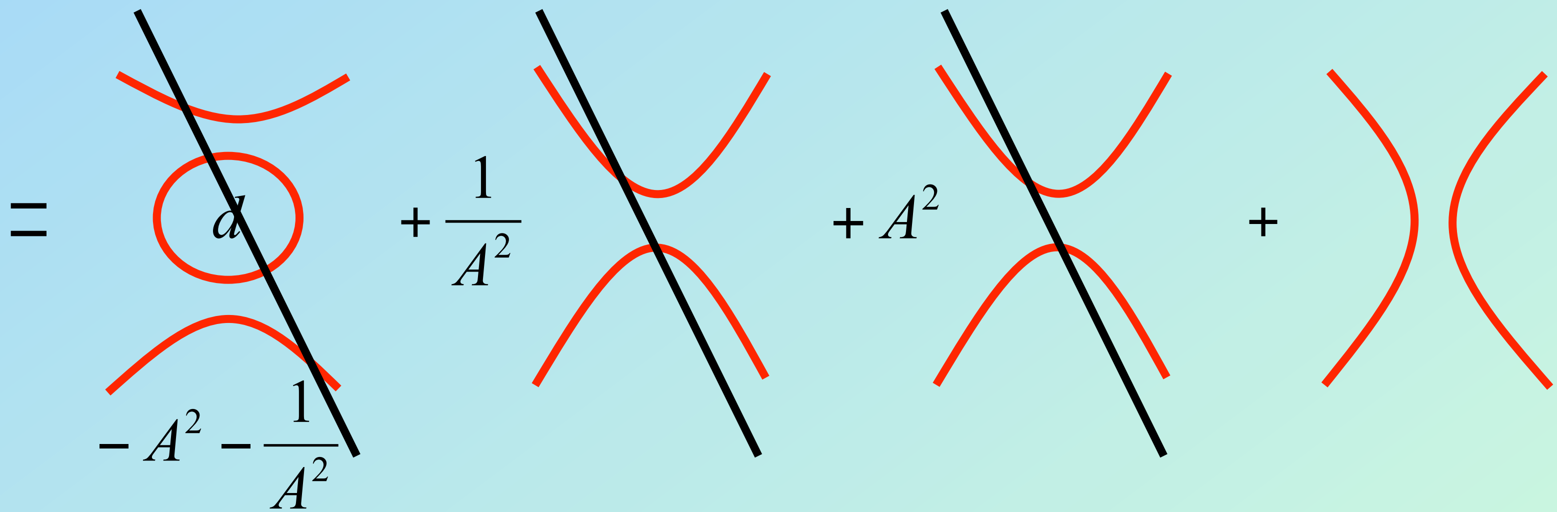
Skein relations



Skein and Reidemeister

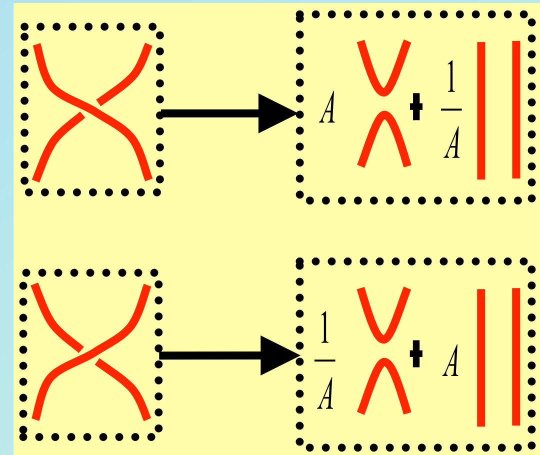




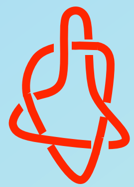


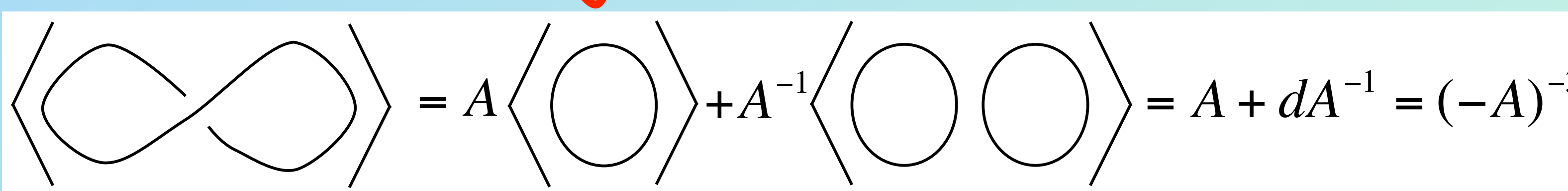
Reidemeister move (II) is satisfied. Similarly (III).

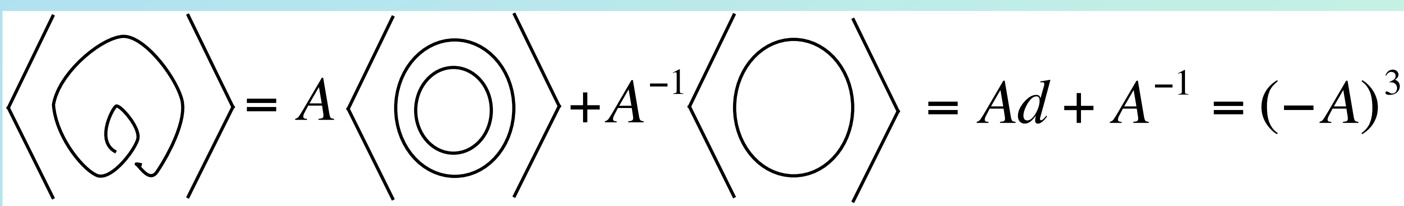
Kauffman bracket

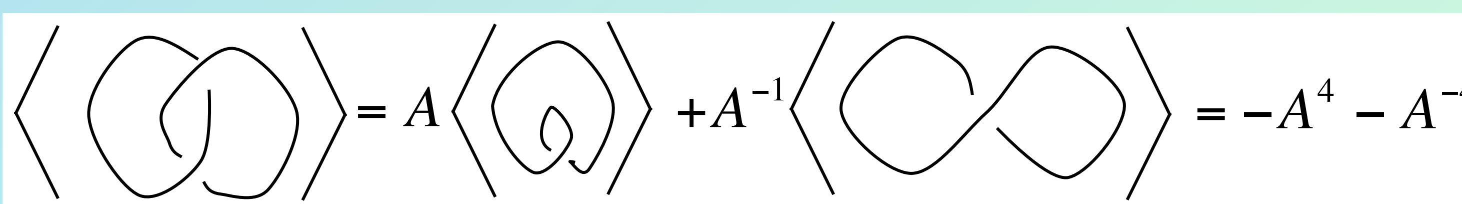


The Skein relations give rise to the **Kauffman bracket**:

Skein() $= \langle L \rangle (A)$

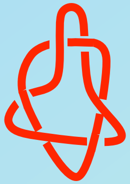






Jones polynomial

The Skein relations give rise to the **Kauffman bracket**:

Skein()= $\langle L \rangle (A)$

To satisfy **move (I)** one needs to define **Jones polynomial**:

$$V_L(A) = (-A)^{3w(L)} \langle L \rangle (A)$$

$w(L)$ is the writhe of link. Easily computable.

Jones polynomials

- If two links have different Jones polynomials then they are inequivalent

=> **use it to distinguish links**

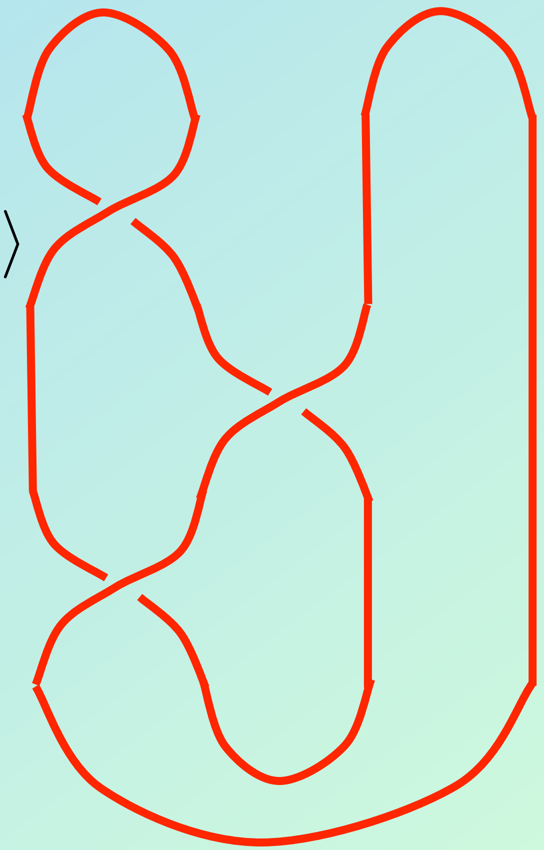
- Jones polynomials keep:

only topological information, **no geometrical**

Jones polynomial from anyons

Braiding evolutions of anyonic states:

$$|\Psi_{\text{final}}\rangle = B_n \ldots B_2 B_1 |\Psi_{\text{initial}}\rangle$$



$$\langle\Psi_{\text{initial}}|\Psi_{\text{final}}\rangle = \langle\Psi_{\text{initial}}|B_n \ldots B_2 B_1|\Psi_{\text{initial}}\rangle$$

$$= \frac{1}{d^{n/2-1}}\langle L(B)\rangle$$

- *Simulate the knot with braiding anyons*
- Translate it to circuit model:
  <=> find trace of matrices

Jones polynomial from QC

Evaluating Jones polynomials is a #P-hard problem.

Belongs to BQP class.

With quantum computers it is **polynomially** easy to *approximate* with additive error.

[Freedman, Kitaev, Larsen, Wang (2002);
Aharonov, Jones, Landau (2005);
et al. Glaser (2009);
Kuperberg (2009)]

Conclusions

Jones polynomials are used for quantum applications:

- **encrypt** quantum information
- **quantum money**
- …

Topological systems that can support **anyons** are currently **engineered**…

http://quantum.leeds.ac.uk/~jiannis