$\cdots \to \mathbf{0} \to \mathbb{Z}^{\mathsf{rx}} \xrightarrow[\mathsf{Sx}]{\partial_2} \mathbb{Z}^{\mathsf{N}} \xrightarrow[\mathsf{Hz}]{\partial_1} \widehat{\mathbb{T}}^{\mathsf{rz}} \cong \mathbb{Z}^{\mathsf{rz}} \to \mathbf{0} \to \cdots$

$\operatorname{coker}\left(R^m \xrightarrow{A} R^n\right) \cong \bigoplus_i R/\alpha_i R$

Quantum error correction with rotors and torsion

Samo Novák
COSMIQ
Inria Paris

Journées C^2
April 2025

Inria

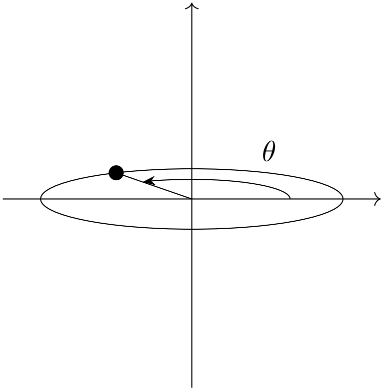
Rotor

We are used to **qubits** (two basis states), and perhaps **qudits** (d basis states). What happens when we send $d \to \infty$ (heuristically) ?

Definition (rotor [VCT23])

A quantum **rotor** is a system whose states we describe using the **circle** group

$$\mathbb{T} := \mathbb{R}/2\pi\mathbb{Z},$$



Rotor

We are used to **qubits** (two basis states), and perhaps **qudits** (d basis states). What happens when we send $d \to \infty$ (heuristically) ?

Definition (rotor [VCT23])

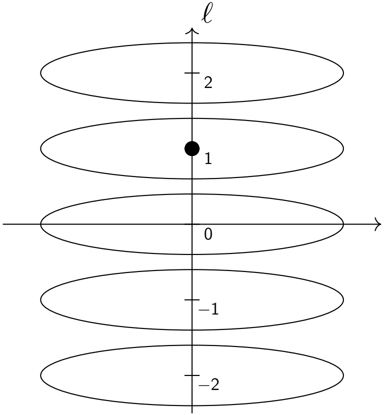
A quantum **rotor** is a system whose states we describe using the **circle** group

$$\mathbb{T} := \mathbb{R}/2\pi\mathbb{Z},$$

and by its **dual**

$$\widehat{\mathbb{T}} \cong \mathbb{Z}.$$

(We won't worry about the duality today.)



Why is this interesting?

Note

From now on, we **focus on one** of these spaces: the one **described by** $\mathbb{Z}$. Results on Pontryagin duality allow this (but we won't go into the details).

Why is this interesting?

Note

From now on, we **focus on one** of these spaces: the one **described by** $\mathbb{Z}$. Results on Pontryagin duality allow this (but we won't go into the details).

If N **qubits**, the state space is $(\mathbb{C}^2)^{\otimes N}$, and we represent it by its **basis** $\mathbb{Z}_2^N$ ($\mathbb{Z}_2 := \mathbb{Z}/2\mathbb{Z}$).

Why is this interesting?

Note

From now on, we **focus on one** of these spaces: the one **described by** $\mathbb{Z}$. Results on Pontryagin duality allow this (but we won't go into the details).

If N **qubits**, the state space is $(\mathbb{C}^2)^{\otimes N}$, and we represent it by its **basis** $\mathbb{Z}_2^N$ ($\mathbb{Z}_2 := \mathbb{Z}/2\mathbb{Z}$).
To study **linear codes**, we use its natural $\mathbb{Z}_2$-**vector space** structure.

Why is this interesting?

Note

From now on, we **focus on one** of these spaces: the one **described by** $\mathbb{Z}$. Results on Pontryagin duality allow this (but we won't go into the details).

If N **qubits**, the state space is $(\mathbb{C}^2)^{\otimes N}$, and we represent it by its **basis** $\mathbb{Z}_2^N$ ($\mathbb{Z}_2 := \mathbb{Z}/2\mathbb{Z}$). To study **linear codes**, we use its natural $\mathbb{Z}_2$-**vector space** structure.

Same idea for **rotors**: N rotors **described by** $\mathbb{Z}^N$.

Why is this interesting?

Note

From now on, we **focus on one** of these spaces: the one **described by** $\mathbb{Z}$. Results on Pontryagin duality allow this (but we won't go into the details).

If N **qubits**, the state space is $(\mathbb{C}^2)^{\otimes N}$, and we represent it by its **basis** $\mathbb{Z}_2^N$ ($\mathbb{Z}_2 := \mathbb{Z}/2\mathbb{Z}$). To study **linear codes**, we use its natural $\mathbb{Z}_2$-**vector space** structure.

Same idea for **rotors**: N rotors **described by** $\mathbb{Z}^N$.

But wait!

Above, $\mathbb{Z}_2$ is a **field** and thus $\mathbb{Z}_2^N$ is a **vector space**. Nice 😎

Why is this interesting?

Note

From now on, we **focus on one** of these spaces: the one **described by** $\mathbb{Z}$. Results on Pontryagin duality allow this (but we won't go into the details).

If N **qubits**, the state space is $(\mathbb{C}^2)^{\otimes N}$, and we represent it by its **basis** $\mathbb{Z}_2^N$ ($\mathbb{Z}_2 := \mathbb{Z}/2\mathbb{Z}$). To study **linear codes**, we use its natural $\mathbb{Z}_2$-**vector space** structure.

Same idea for **rotors**: N rotors **described by** $\mathbb{Z}^N$.

But wait!

Above, $\mathbb{Z}_2$ is a **field** and thus $\mathbb{Z}_2^N$ is a **vector space**. Nice 😎

However, the ring $\mathbb{Z}$ is **not** a field, and $\mathbb{Z}^N$ is **not** a vector space!!!

Why is this interesting?

Note

From now on, we **focus on one** of these spaces: the one **described by** $\mathbb{Z}$. Results on Pontryagin duality allow this (but we won't go into the details).

If N **qubits**, the state space is $(\mathbb{C}^2)^{\otimes N}$, and we represent it by its **basis** $\mathbb{Z}_2^N$ ($\mathbb{Z}_2 := \mathbb{Z}/2\mathbb{Z}$).
To study **linear codes**, we use its natural $\mathbb{Z}_2$-**vector space** structure.

Same idea for **rotors**: N rotors **described by** $\mathbb{Z}^N$.

But wait!

Above, $\mathbb{Z}_2$ is a **field** and thus $\mathbb{Z}_2^N$ is a **vector space**. Nice 😎

However, the ring $\mathbb{Z}$ is **not** a field, and $\mathbb{Z}^N$ is **not** a vector space!!!
Instead, $\mathbb{Z}^N$ is a $\mathbb{Z}$-**module**.

Why is this interesting?

Note

From now on, we **focus on one** of these spaces: the one **described by** $\mathbb{Z}$. Results on Pontryagin duality allow this (but we won't go into the details).

If N **qubits**, the state space is $(\mathbb{C}^2)^{\otimes N}$, and we represent it by its **basis** $\mathbb{Z}_2^N$ ($\mathbb{Z}_2 := \mathbb{Z}/2\mathbb{Z}$).
To study **linear codes**, we use its natural $\mathbb{Z}_2$-**vector space** structure.

Same idea for **rotors**: N rotors **described by** $\mathbb{Z}^N$.

But wait!

Above, $\mathbb{Z}_2$ is a **field** and thus $\mathbb{Z}_2^N$ is a **vector space**. Nice 😎

However, the ring $\mathbb{Z}$ is **not** a field, and $\mathbb{Z}^N$ is **not** a vector space!!!
Instead, $\mathbb{Z}^N$ is a $\mathbb{Z}$-**module**.

Conclusion: need **more general** machinery, unexpected phenomena happen: **torsion**.

Okay whatever. Why is this REALLY interesting?

Motivation: Building qubit systems from rotors

This is something we **can build** and use, so it's worth studying what it can do. In particular, we can use physical **rotors** to **encode other kinds** of systems (e.g. **qubits**).

Okay whatever. Why is this REALLY interesting?

Motivation: Building qubit systems from rotors

This is something we **can build** and use, so it's worth studying what it can do. In particular, we can use physical **rotors** to **encode other kinds** of systems (e.g. **qubits**).

Motivation: Revealing subtleties

The more general machinery reveals **subtle details** that are also present, but **hidden**, in more **conventional** constructions.

Escape-velocity intro to module theory

Let R be a ring. Examples: $\mathbb{Z}$, $\mathbb{Z}_{42} := {}^{\mathbb{Z}}\!/_{42\mathbb{Z}}$, $\mathbb{R}$ (field).

Escape-velocity intro to module theory

Let R be a ring. Examples: $\mathbb{Z}$, $\mathbb{Z}_{42} := {}^{\mathbb{Z}}\!/_{42\mathbb{Z}}$, $\mathbb{R}$ (field).

Definition (module)

An R-**module** M is an **abelian group** with a compatible **action** of R (**scalar multiplication**).

Generalizes vector spaces: a **vector space** is by definition a **module over a field**.

Escape-velocity intro to module theory

Let R be a ring. Examples: $\mathbb{Z}$, $\mathbb{Z}_{42} := \mathbb{Z}/42\mathbb{Z}$, $\mathbb{R}$ (field).

Definition (module)

An R-**module** M is an **abelian group** with a compatible **action** of R (**scalar multiplication**).

Generalizes vector spaces: a **vector space** is by definition a **module over a field**.

Idea

If R is **not a field**, sometimes **weird stuff** happens.

Torsion (algebra)

In this talk, we **take** $R = \mathbb{Z}$, which is a PID (a nice kind of ring).

Torsion (algebra)

In this talk, we **take** $R = \mathbb{Z}$, which is a PID (a nice kind of ring).

Definition (torsion; simplified)

Let M be a $\mathbb{Z}$-module. If there is $\underline{v} \in M$ and a **nonzero** $n \in \mathbb{Z}$, such that

$$n \cdot \underline{v} = \underline{0}$$

then $\underline{v}$ is a **torsion** element. The set of such $\underline{v}$ is a **torsion submodule**.

Torsion (algebra)

In this talk, we **take** $R = \mathbb{Z}$, which is a PID (a nice kind of ring).

Definition (torsion; simplified)

Let M be a $\mathbb{Z}$-module. If there is $\underline{v} \in M$ and a **nonzero** $n \in \mathbb{Z}$, such that

$$n \cdot \underline{v} = \underline{0}$$

then $\underline{v}$ is a **torsion** element. The set of such $\underline{v}$ is a **torsion submodule**.

Example

Let $M = \mathbb{Z}_2 = {}^{\mathbb{Z}}\!/_{2\mathbb{Z}} = \{\bar{0}, \bar{1}\}$ as $\mathbb{Z}$-module. Then:

$$4 \cdot \bar{1} = \overline{4 \cdot 1 \pmod 2} = \bar{0}.$$

The whole $\mathbb{Z}_2$ is a torsion $\mathbb{Z}$-module!

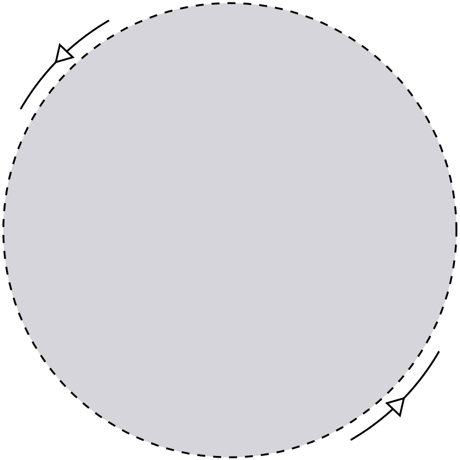
Torsion (topology): Example of $\mathbb{RP}^2$

Torsion describes a kind of **weirdness** of **topological spaces**.

Torsion (topology): Example of $\mathbb{RP}^2$

Torsion describes a kind of **weirdness** of **topological spaces**.

Right: real **projective plane** $\mathbb{RP}^2$ = **disk**, but **glue opposite** points of the boundary.



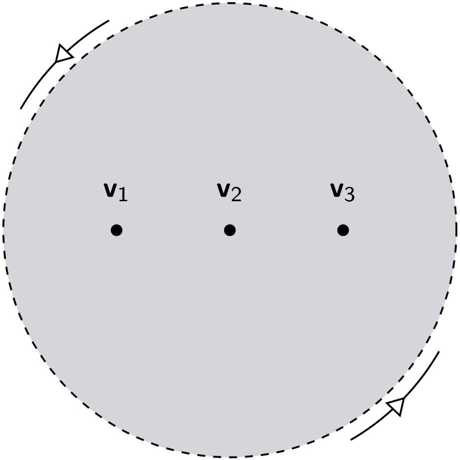
Torsion (topology): Example of $\mathbb{RP}^2$

Torsion describes a kind of **weirdness** of **topological spaces**.

Right: real **projective plane** $\mathbb{RP}^2 =$ **disk**, but **glue opposite** points of the boundary.

Cellulation with

- **points** $\{\mathbf{v}_1, \ldots, \mathbf{v}_3\}$,



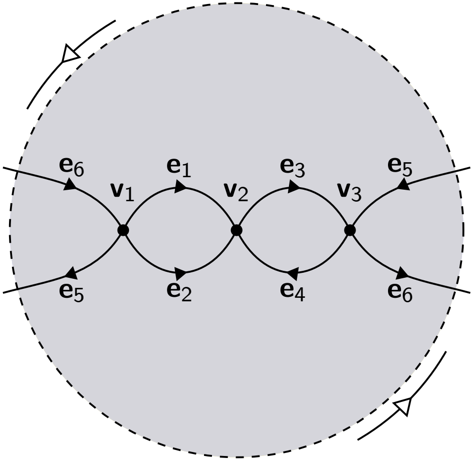
Torsion (topology): Example of $\mathbb{RP}^2$

Torsion describes a kind of **weirdness** of **topological spaces**.

Right: real **projective plane** $\mathbb{RP}^2$ = **disk**, but **glue opposite** points of the boundary.

Cellulation with

- **points** $\{\mathbf{v}_1, \ldots, \mathbf{v}_3\}$,
- oriented **edges** $\{\mathbf{e}_1, \ldots, \mathbf{e}_6\}$,



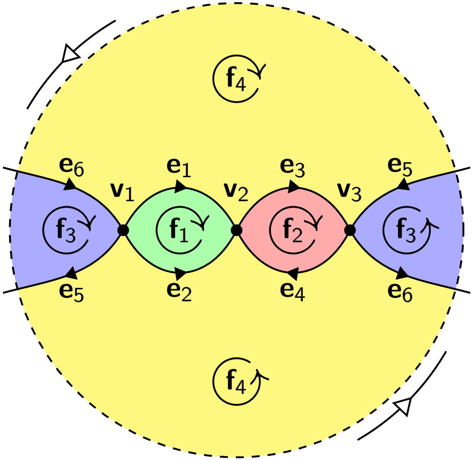
Torsion (topology): Example of $\mathbb{RP}^2$

Torsion describes a kind of **weirdness** of **topological spaces**.

Right: real **projective plane** $\mathbb{RP}^2 =$ **disk**, but **glue opposite** points of the boundary.

Cellulation with

- **points** $\{\mathbf{v}_1, \ldots, \mathbf{v}_3\}$,
- oriented **edges** $\{\mathbf{e}_1, \ldots, \mathbf{e}_6\}$,
- oriented **faces** $\{\mathbf{f}_1, \ldots, \mathbf{f}_4\}$.



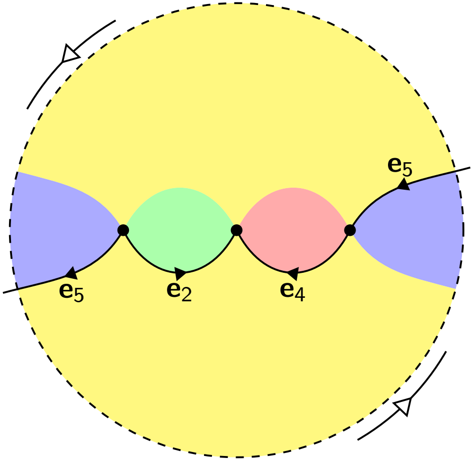
Torsion (topology): Example of $\mathbb{RP}^2$

Torsion describes a kind of **weirdness** of **topological spaces**.

Right: real **projective plane** $\mathbb{RP}^2$ = **disk**, but **glue opposite** points of the boundary.

Cellulation with

- **points** $\{\mathbf{v}_1, \ldots, \mathbf{v}_3\}$,
- oriented **edges** $\{\mathbf{e}_1, \ldots, \mathbf{e}_6\}$,
- oriented **faces** $\{\mathbf{f}_1, \ldots, \mathbf{f}_4\}$.



Intuition

Torsion, here $\mathbb{Z}_2$, corresponds to a **cycle** which must be **traversed multiple times** (2) to **come back** exactly to the **same point** in the **same orientation**.

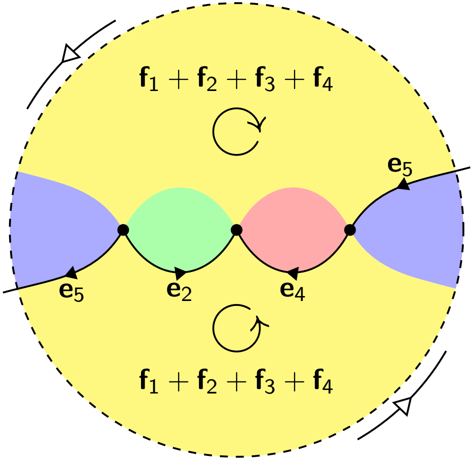
Torsion (topology): Example of $\mathbb{RP}^2$

Torsion describes a kind of **weirdness** of **topological spaces**.

Right: real **projective plane** $\mathbb{RP}^2$ = **disk**, but **glue opposite** points of the boundary.

Cellulation with

- **points** $\{\mathbf{v}_1, \ldots, \mathbf{v}_3\}$,
- oriented **edges** $\{\mathbf{e}_1, \ldots, \mathbf{e}_6\}$,
- oriented **faces** $\{\mathbf{f}_1, \ldots, \mathbf{f}_4\}$.



Note *linear* structure!

More precisely

Cycle $\underline{x} = -\mathbf{e}_2 + \mathbf{e}_4 + \mathbf{e}_5$ is **not a boundary** of a face. But its **multiple** $2\underline{x}$ is the boundary of $\mathbf{f}_1 + \mathbf{f}_2 + \mathbf{f}_3 + \mathbf{f}_4$.
Thus $\mathbb{Z}_2$ **torsion**.

Chain complex: the previous idea in more detail

A way to construct a code is to **cellulate** a topological **space** and build a **chain complex**:

Definition (chain complex CSS code)

A **chain complex** representing a **CSS code** is:

$$C_2 = \mathbb{Z}^{r_X} \xrightarrow{\partial_2 = H_X^\top} C_1 = \mathbb{Z}^N \xrightarrow{\partial_1 = H_Z} C_0 = \mathbb{Z}^{r_Z}$$

where $\operatorname{im} \partial_2 \subseteq \ker \partial_1$.

Chain complex: the previous idea in more detail

A way to construct a code is to **cellulate** a topological **space** and build a **chain complex**:

Definition (chain complex CSS code)

A **chain complex** representing a **CSS code** is:

$$C_2 = \mathbb{Z}^{r_X} \xrightarrow{\partial_2 = H_X^\top} C_1 = \mathbb{Z}^N \xrightarrow{\partial_1 = H_Z} C_0 = \mathbb{Z}^{r_Z}$$

where $\operatorname{im}\partial_2 \subseteq \ker\partial_1$. The spaces are:

- $C_2 = \langle \mathbf{f}_1, \ldots, \mathbf{f}_{r_X} \rangle_{\mathbb{Z}}$ spanned by **faces**, corresponding to X-**stabilizers**,

Chain complex: the previous idea in more detail

A way to construct a code is to **cellulate** a topological **space** and build a **chain complex**:

Definition (chain complex CSS code)

A **chain complex** representing a **CSS code** is:

$$C_2 = \mathbb{Z}^{r_X} \xrightarrow{\partial_2 = H_X^\top} C_1 = \mathbb{Z}^N \xrightarrow{\partial_1 = H_Z} C_0 = \mathbb{Z}^{r_Z}$$

where $\operatorname{im} \partial_2 \subseteq \ker \partial_1$. The spaces are:

- $C_2 = \langle \mathbf{f}_1, \ldots, \mathbf{f}_{r_X} \rangle_{\mathbb{Z}}$ spanned by **faces**, corresponding to X-**stabilizers**,
- $C_1 = \langle \mathbf{e}_1, \ldots, \mathbf{e}_N \rangle_{\mathbb{Z}}$ spanned by **edges**, corresponding to **physical rotors** (or qubits),

Chain complex: the previous idea in more detail

A way to construct a code is to **cellulate** a topological **space** and build a **chain complex**:

Definition (chain complex CSS code)

A **chain complex** representing a **CSS code** is:

$$C_2 = \mathbb{Z}^{r_X} \xrightarrow{\partial_2 = H_X^\top} C_1 = \mathbb{Z}^N \xrightarrow{\partial_1 = H_Z} C_0 = \mathbb{Z}^{r_Z}$$

where $\operatorname{im} \partial_2 \subseteq \ker \partial_1$. The spaces are:

- $C_2 = \langle \mathbf{f}_1, \ldots, \mathbf{f}_{r_X} \rangle_{\mathbb{Z}}$ spanned by **faces**, corresponding to X-**stabilizers**,
- $C_1 = \langle \mathbf{e}_1, \ldots, \mathbf{e}_N \rangle_{\mathbb{Z}}$ spanned by **edges**, corresponding to **physical rotors** (or qubits),
- $C_0 = \langle \mathbf{v}_1, \ldots, \mathbf{v}_{r_Z} \rangle_{\mathbb{Z}}$ spanned by **vertices**, corresponding to Z-**syndromes**.

Chain complex: the previous idea in more detail

A way to construct a code is to **cellulate** a topological **space** and build a **chain complex**:

Definition (chain complex CSS code)

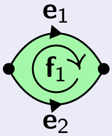
A **chain complex** representing a **CSS code** is:

$$C_2 = \mathbb{Z}^{r_X} \xrightarrow{\partial_2 = H_X^\top} C_1 = \mathbb{Z}^N \xrightarrow{\partial_1 = H_Z} C_0 = \mathbb{Z}^{r_Z}$$

where $\operatorname{im}\partial_2 \subseteq \ker \partial_1$. The spaces are:

- $C_2 = \langle \mathbf{f}_1, \ldots, \mathbf{f}_{r_X} \rangle_{\mathbb{Z}}$ spanned by **faces**, corresponding to X-**stabilizers**,
- $C_1 = \langle \mathbf{e}_1, \ldots, \mathbf{e}_N \rangle_{\mathbb{Z}}$ spanned by **edges**, corresponding to **physical rotors** (or qubits),
- $C_0 = \langle \mathbf{v}_1, \ldots, \mathbf{v}_{r_Z} \rangle_{\mathbb{Z}}$ spanned by **vertices**, corresponding to Z-**syndromes**.

The **maps** ∂_n describe the **incidence** of n-dim cells on their $(n-1)$-dim **boundaries**, e.g. $\partial_2(\mathbf{f}_1) = +\mathbf{e}_1 - \mathbf{e}_2$.



Chain complex: the previous idea in more detail

A way to construct a code is to **cellulate** a topological **space** and build a **chain complex**:

Definition (chain complex CSS code)

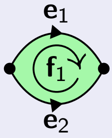
A **chain complex** representing a **CSS code** is:

$$C_2 = \mathbb{Z}^{r_X} \xrightarrow{\partial_2 = H_X^\top} C_1 = \mathbb{Z}^N \xrightarrow{\partial_1 = H_Z} C_0 = \mathbb{Z}^{r_Z}$$

where $\operatorname{im}\partial_2 \subseteq \ker\partial_1$. The spaces are:

- $C_2 = \langle \mathbf{f}_1, \ldots, \mathbf{f}_{r_X} \rangle_{\mathbb{Z}}$ spanned by **faces**, corresponding to X-**stabilizers**,
- $C_1 = \langle \mathbf{e}_1, \ldots, \mathbf{e}_N \rangle_{\mathbb{Z}}$ spanned by **edges**, corresponding to **physical rotors** (or qubits),
- $C_0 = \langle \mathbf{v}_1, \ldots, \mathbf{v}_{r_Z} \rangle_{\mathbb{Z}}$ spanned by **vertices**, corresponding to Z-**syndromes**.

The **maps** ∂_n describe the **incidence** of n-dim cells on their $(n-1)$-dim **boundaries**, e.g. $\partial_2(\mathbf{f}_1) = +\mathbf{e}_1 - \mathbf{e}_2$.

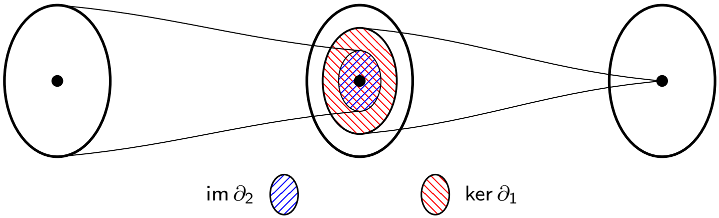


These boundary maps give **parity check matrices** H_X and H_Z.

Encoded logical space: homology

In a **chain complex** representing a CSS code

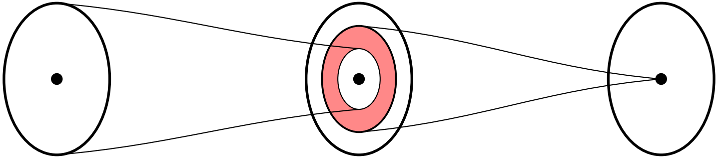
$$C_2 = \mathbb{Z}^{r_X} \xrightarrow{\partial_2 = H_X^\top} C_1 = \mathbb{Z}^N \xrightarrow{\partial_1 = H_Z} C_0 = \mathbb{Z}^{r_Z}$$



Encoded logical space: homology

In a **chain complex** representing a CSS code

$$C_2 = \mathbb{Z}^{r_X} \xrightarrow{\partial_2 = H_X^\top} C_1 = \mathbb{Z}^N \xrightarrow{\partial_1 = H_Z} C_0 = \mathbb{Z}^{r_Z}$$



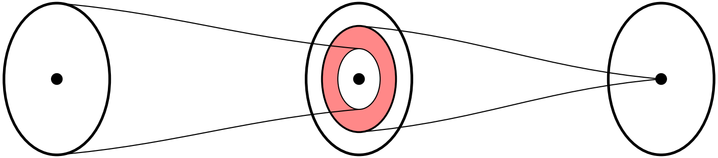
the **logical operators** are found in the first **homology** module:

$$H_1 := {}^{\ker \partial_1}\!/\!_{\operatorname{im} \partial_2}.$$

Encoded logical space: homology

In a **chain complex** representing a CSS code

$$C_2 = \mathbb{Z}^{r_X} \xrightarrow{\partial_2 = H_X^\top} C_1 = \mathbb{Z}^N \xrightarrow{\partial_1 = H_Z} C_0 = \mathbb{Z}^{r_Z}$$



the **logical operators** are found in the first **homology** module:

$$H_1 := {}^{\ker \partial_1}\!/_{\operatorname{im} \partial_2}.$$

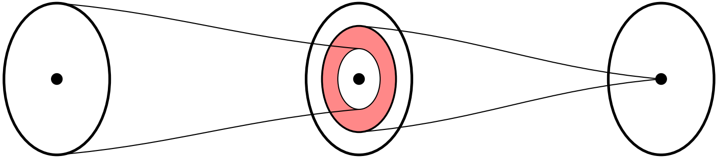
Recall example

Cycle $\underline{x} = -\mathbf{e}_2 + \mathbf{e}_4 + \mathbf{e}_5$ is **not a boundary** of a face. But its **multiple** $2\underline{x}$ is. Thus $\mathbb{Z}_2$ torsion.

Encoded logical space: homology

In a **chain complex** representing a CSS code

$$C_2 = \mathbb{Z}^{r_X} \xrightarrow{\partial_2 = H_X^\top} C_1 = \mathbb{Z}^N \xrightarrow{\partial_1 = H_Z} C_0 = \mathbb{Z}^{r_Z}$$



the **logical operators** are found in the first **homology** module:

$$H_1 := {}^{\ker \partial_1}\!/\!_{\operatorname{im} \partial_2}.$$

Key idea

Torsion comes from a **cycle** of edges that is **not a boundary**, but its **multiple is**.

This is given by the **image** $\operatorname{im} \partial_2$.

What does torsion mean?

In the $\mathbb{RP}^2$ example, the **homology** is $H_1(\mathbb{RP}^2) = \mathbb{Z}_2$, even though it is constructed from rotors ($\mathbb{Z}$). This is **torsion**.

What does torsion mean?

In the $\mathbb{RP}^2$ example, the **homology** is $H_1(\mathbb{RP}^2) = \mathbb{Z}_2$, even though it is constructed from rotors ($\mathbb{Z}$). This is **torsion**.

The $\mathbb{Z}_2$ means our **rotor** system **encodes a qubit**.

What does torsion mean?

In the $\mathbb{RP}^2$ example, the **homology** is $H_1(\mathbb{RP}^2) = \mathbb{Z}_2$, even though it is constructed from rotors ($\mathbb{Z}$). This is **torsion**.

The $\mathbb{Z}_2$ means our **rotor** system **encodes a qubit**.

Key takeaway

In the **general** setting of **rotors**, we can use one kind of system to **encode another kind**.

What does torsion mean?

In the $\mathbb{RP}^2$ example, the **homology** is $H_1(\mathbb{RP}^2) = \mathbb{Z}_2$, even though it is constructed from rotors ($\mathbb{Z}$). This is **torsion**.

The $\mathbb{Z}_2$ means our **rotor** system **encodes a qubit**.

Key takeaway

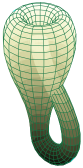
In the **general** setting of **rotors**, we can use one kind of system to **encode another kind**.

Example: Klein bottle

A **Klein bottle** $\mathbb{K}^2$ has homology

$$H_1(\mathbb{K}^2) = \mathbb{Z} \oplus \mathbb{Z}_2.$$

A rotor code defined on a cellulation of $\mathbb{K}^2$ encodes a **rotor** and a **qubit**.



[Ttt06]

What does torsion mean?

In the $\mathbb{RP}^2$ example, the **homology** is $H_1(\mathbb{RP}^2) = \mathbb{Z}_2$, even though it is constructed from rotors ($\mathbb{Z}$). This is **torsion**.

The $\mathbb{Z}_2$ means our **rotor** system **encodes a qubit**.

Key takeaway

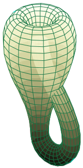
In the **general** setting of **rotors**, we can use one kind of system to **encode another kind**.

Example: Klein bottle

A **Klein bottle** $\mathbb{K}^2$ has homology

$$H_1(\mathbb{K}^2) = \mathbb{Z} \oplus \mathbb{Z}_2.$$

A rotor code defined on a cellulation of $\mathbb{K}^2$ encodes a **rotor** and a **qubit**.



[Ttt06]

Key observation

We can obtain **mixed-dimension** systems.

$\cdots \to \mathbf{0} \to \mathbb{Z}^{r_x} \xrightarrow[S_x]{\partial_2} \mathbb{Z}^N \xrightarrow[H_z]{\partial_1} \widehat{\mathbb{T}}^{r_z} \cong \mathbb{Z}^{r_z} \to \mathbf{0} \to \cdots$ $\quad \operatorname{coker}(\mathbf{\cdot} \xrightarrow{A} R^n) \cong \bigoplus_i R/\alpha_i R$

Thank you for your attention!

Quantum error correction with rotors and torsion

Samo Novák
COSMIQ
Inria Paris

Journées C^2
April 2025

Bibliography follows.

Bibliography

- Tttrung, *Klein bottle made with gnuplot 4.0.*, https://commons.wikimedia.org/wiki/File:Klein_bottle.svg, July 2006.
- Christophe Vuillot, Alessandro Ciani, and Barbara M. Terhal, *Homological Quantum Rotor Codes: Logical Qubits from Torsion*, September 2023.