

# Restrictions on Transversal Encoded Quantum Gate Sets

Bryan Eastin and Emanuel Knill

National Institute of Standards and Technology  
Boulder, CO

February 2008

# Motivation

Transversal encoded gates are inherently fault tolerant.

## Desired

A quantum code with a universal, transversal encoded gate set

Universal, transversal encoded gate sets are hard to find.

## Alternate desire

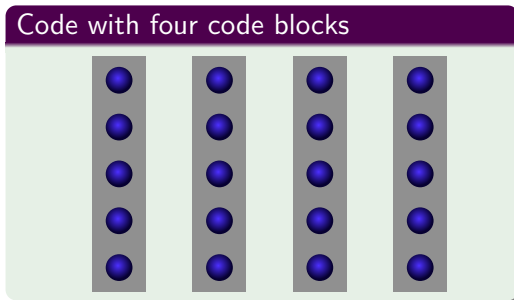
A proof that such gate sets don't exist

## Previous work

- Qubit stabilizer codes:  
Zeng, Cross, and Chuang, arXiv:0706.1382
- Qudit stabilizer codes:  
Chen, Chung, Cross, Zeng, and Chuang, arXiv:0801.2360

# Definitions

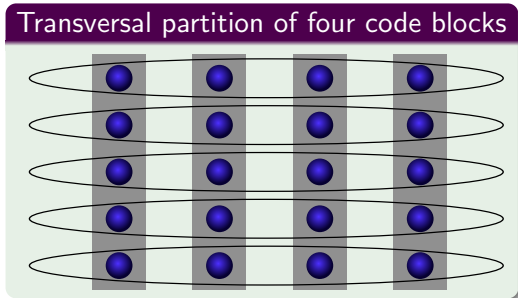
- Quantum code** Subspace of Hilbert space, defined by projector,  $P$
- Detectable error** Error  $E$  satisfying  $PEP \propto P$
- Code block** Unit of independent error detection



- Transversal partition** Partition such that each part contains one subsystem from each code block
- Transversal operator** Operator which only couples subsystems within a transversal partition

# Definitions

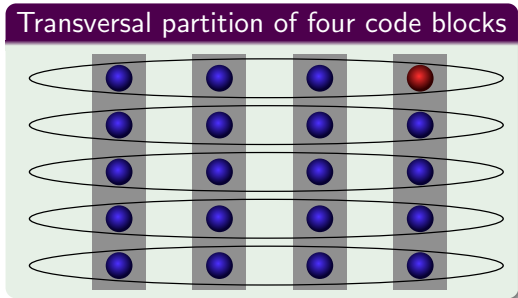
- Quantum code** Subspace of Hilbert space, defined by projector,  $P$
- Detectable error** Error  $E$  satisfying  $PEP \propto P$
- Code block** Unit of independent error detection



- Transversal partition** Partition such that each part contains one subsystem from each code block
- Transversal operator** Operator which only couples subsystems within a transversal partition

# Definitions

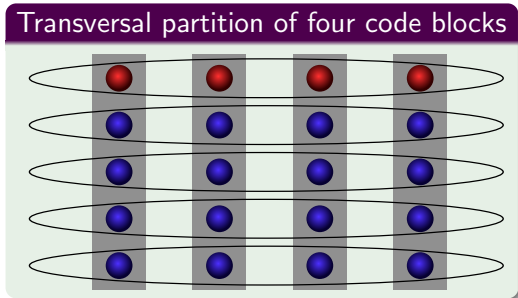
- Quantum code** Subspace of Hilbert space, defined by projector,  $P$
- Detectable error** Error  $E$  satisfying  $PEP \propto P$
- Code block** Unit of independent error detection



- Transversal partition** Partition such that each part contains one subsystem from each code block
- Transversal operator** Operator which only couples subsystems within a transversal partition

# Definitions

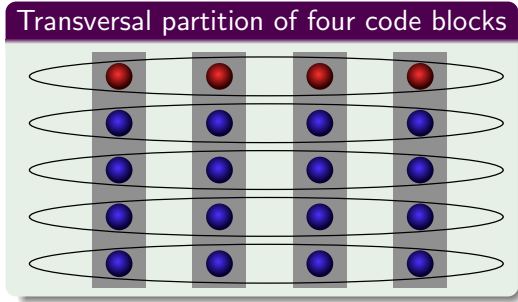
- Quantum code** Subspace of Hilbert space, defined by projector,  $P$
- Detectable error** Error  $E$  satisfying  $PEP \propto P$
- Code block** Unit of independent error detection



- Transversal partition** Partition such that each part contains one subsystem from each code block
- Transversal operator** Operator which only couples subsystems within a transversal partition

# Definitions

**Code block** Unit of independent error detection



**Transversal partition** Partition such that each part contains one subsystem from each code block

**Transversal operator** Operator which only couples subsystems within a transversal partition

**Product operator** Operator which does not couple subsystems

# Inspiration

Consider a code with logical gates of the form

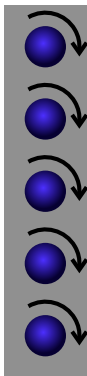
$$\overline{R(Z, \theta)} = e^{i\theta Z} \otimes e^{i\theta Z} \otimes \dots \otimes e^{i\theta Z} = \prod_j e^{i\theta Z_j} .$$

For small  $\theta$ ,

$$\overline{R(Z, \theta)} \approx \prod_j (I + i\theta Z_j) \approx I + i\theta \sum_j Z_j .$$

$Z_j$  must be a logical gate. Else,  $\overline{R(Z, \theta)}$  would not be one.

Conclusion: Such codes cannot be local-error-detecting.



# Inspiration Issues

Consider a code with logical gates of the form

$$\overline{R(Z, \theta)} = e^{i\theta Z} \otimes e^{i\theta Z} \otimes \dots \otimes e^{i\theta Z} = \prod_j e^{i\theta Z_j} .$$

The transversal logical gates need not have this form.

For small  $\theta$ ,

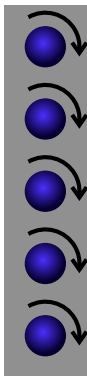
$$\overline{R(Z, \theta)} \approx \prod_j (I + i\theta Z_j) \approx I + i\theta \sum_j Z_j .$$

Requires continuity.

$Z_j$  must be a logical gate. Else,  $\overline{R(Z, \theta)}$  would not be one.

True if the code is additive.

Conclusion: Such codes cannot be local-error-detecting.

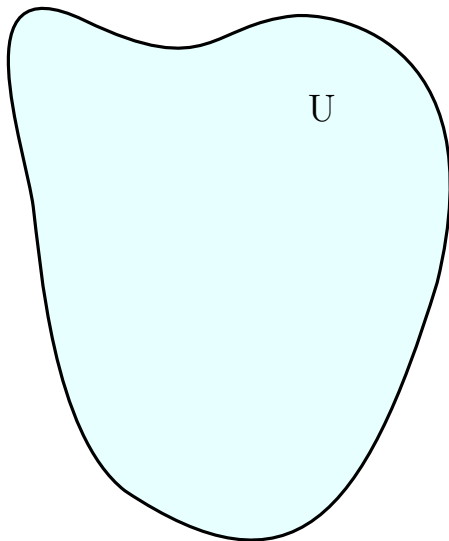


# Outline of the Proof

## Steps

- 1 The logical product operators form a Lie subgroup  $G$ .
- 2  $G$  partitions into a finite number of cosets.
- 3 Each coset of  $G$  yields one logically distinct operator.
- 4 A finite set of operators is not universal. Product operators are not universal.
- 5 The transversal logical operators are not universal.

# The logical product operators are a Lie subgroup



U - Unitary operators

T - Product operators,  $T$  such that

$$T = \bigotimes_j U_j$$

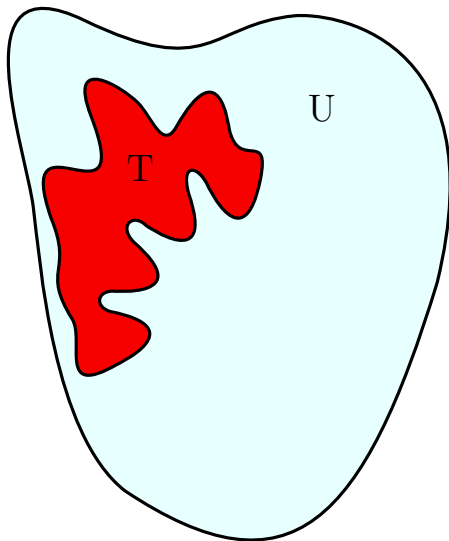
L - Logical operators,  $L$  such that

$$(I - P)LP = 0$$

$G = L \cap T$  - Logical product operators

$G$  is a Lie subgroup.

# The logical product operators are a Lie subgroup



U - Unitary operators

T - Product operators,  $T$  such that

$$T = \bigotimes_j U_j$$

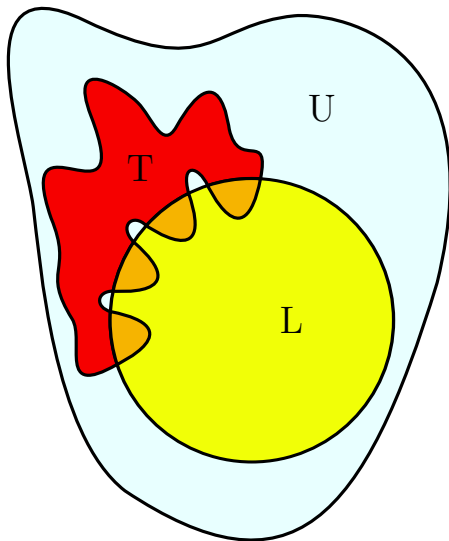
L - Logical operators,  $L$  such that

$$(I - P)LP = 0$$

$G = L \cap T$  - Logical product operators

$G$  is a Lie subgroup.

# The logical product operators are a Lie subgroup



U - Unitary operators

T - Product operators,  $T$  such that

$$T = \bigotimes_j U_j$$

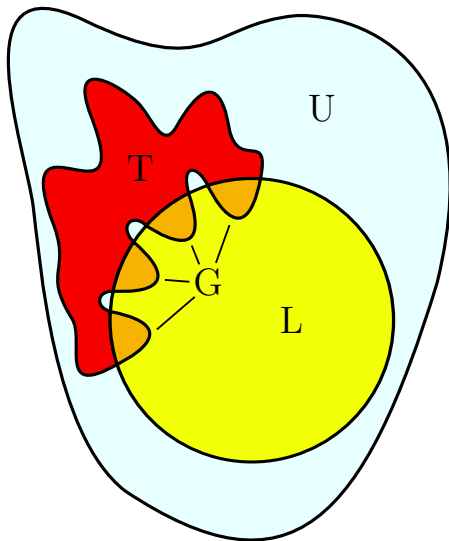
L - Logical operators,  $L$  such that

$$(I - P)LP = 0$$

$G = L \cap T$  - Logical product operators

$G$  is a Lie subgroup.

# The logical product operators are a Lie subgroup



U - Unitary operators

T - Product operators,  $T$  such that

$$T = \bigotimes_j U_j$$

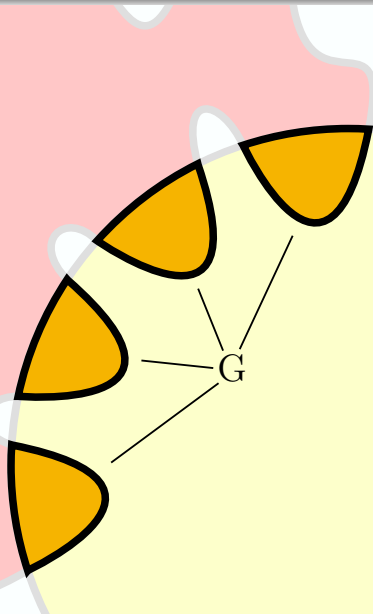
L - Logical operators,  $L$  such that

$$(I - P)LP = 0$$

$G = L \cap T$  - Logical product operators

$G$  is a Lie subgroup.

# Partitioning the logical product operators



Because  $G$  is a Lie group,

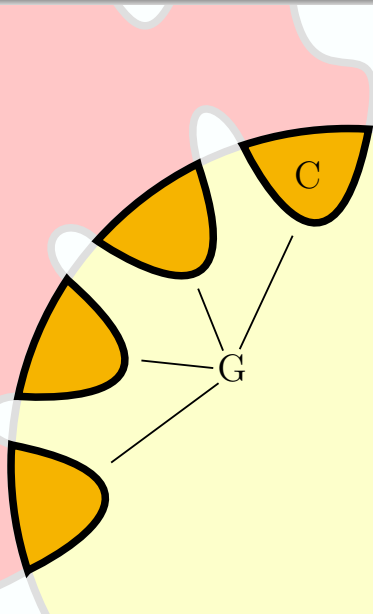
- $G$  can be partitioned into connected components
- one is the connected component of the identity,  $C$ , and a Lie group
- all other components are cosets of  $C$

$F$  - Set of representatives of the cosets

The cosets are

- discrete
- finite in number

# Partitioning the logical product operators



Because  $G$  is a Lie group,

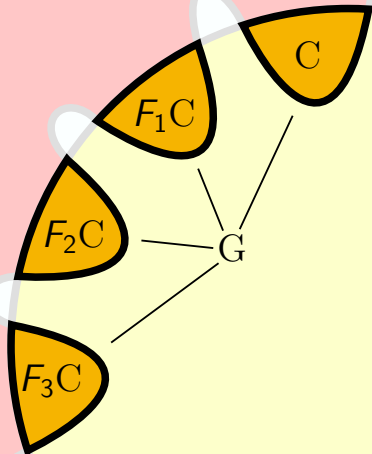
- $G$  can be partitioned into connected components
- one is the connected component of the identity,  $C$ , and a Lie group
- all other components are cosets of  $C$

$F$  - Set of representatives of the cosets

The cosets are

- discrete
- finite in number

# Partitioning the logical product operators



Because  $G$  is a Lie group,

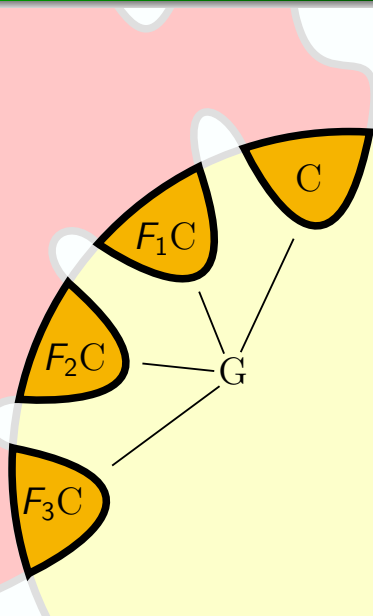
- $G$  can be partitioned into connected components
- one is the connected component of the identity,  $C$ , and a Lie group
- all other components are cosets of  $C$

$F$  - Set of representatives of the cosets

The cosets are

- discrete
- finite in number

# Each coset yields one logically distinct operator



The component of the identity,  $C$ ,

- is a Lie subgroup of  $T$
- has a Lie algebra of sums of local, Hermitian operators

A local, Hermitian operator  $H$  satisfies the local-error-detection condition:

$$PHP \propto P.$$

Elements of the Lie algebra of  $C$

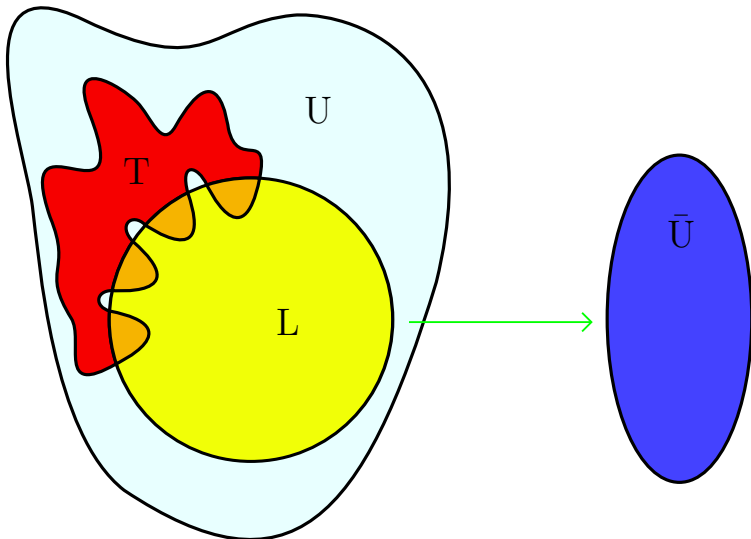
- act trivially on the code space
- are logical operators

Elements in  $C$  implement logical identity.

$F$  indexes the logically distinct gates in  $G$ .

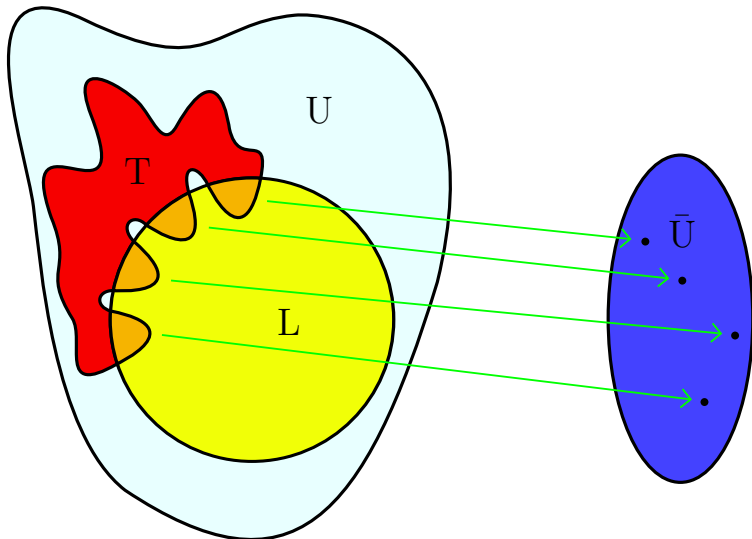
# A finite number of operators is not universal

Consider the space of operators on the encoded system.



# A finite number of operators is not universal

Consider the space of operators on the encoded system.



# A finite number of operators is not universal

Consider the space of operators on the encoded system.

The logically distinct operators in  $\mathcal{G}$  are

- represented by  $F$
- finite in number

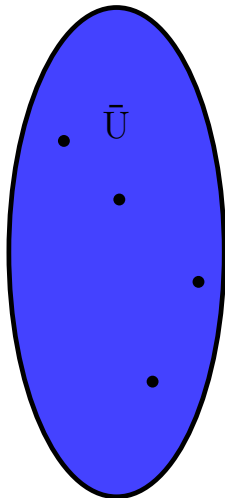
Desired set of logically distinct operators is infinite.

Arbitrary approximation is impossible.

## Theorem 1

A local-error-detecting code cannot have a universal set of product operators.

Contrasts with a finite **basis** of gates



# A finite number of operators is not universal

Consider the space of operators on the encoded system.

The logically distinct operators in  $\mathcal{G}$  are

- represented by  $F$
- finite in number

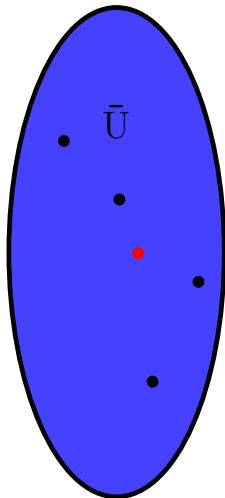
Desired set of logically distinct operators is infinite.

Arbitrary approximation is impossible.

## Theorem 1

A local-error-detecting code cannot have a universal set of product operators.

Contrasts with a finite **basis** of gates



# A finite number of operators is not universal

Consider the space of operators on the encoded system.

The logically distinct operators in  $\mathcal{G}$  are

- represented by  $F$
- finite in number

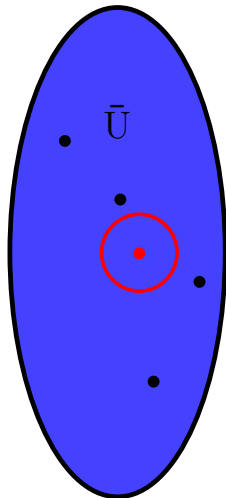
Desired set of logically distinct operators is infinite.

Arbitrary approximation is impossible.

## Theorem 1

A local-error-detecting code cannot have a universal set of product operators.

Contrasts with a finite **basis** of gates



# A finite number of operators is not universal

Consider the space of operators on the encoded system.

The logically distinct operators in  $\mathcal{G}$  are

- represented by  $F$
- finite in number

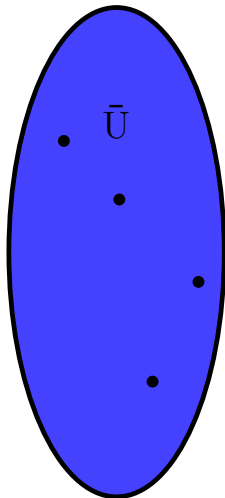
Desired set of logically distinct operators is infinite.

Arbitrary approximation is impossible.

## Theorem 1

A local-error-detecting code cannot have a universal set of product operators.

Contrasts with a finite **basis** of gates



# A finite number of operators is not universal

Consider the space of operators on the encoded system.

The logically distinct operators in  $\mathcal{G}$  are

- represented by  $F$
- finite in number

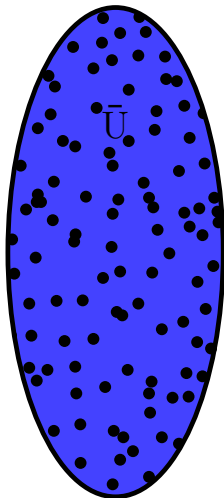
Desired set of logically distinct operators is infinite.

Arbitrary approximation is impossible.

## Theorem 1

A local-error-detecting code cannot have a universal set of product operators.

Contrasts with a finite **basis** of gates



# A finite number of operators is not universal

Consider the space of operators on the encoded system.

The logically distinct operators in  $\mathcal{G}$  are

- represented by  $F$
- finite in number

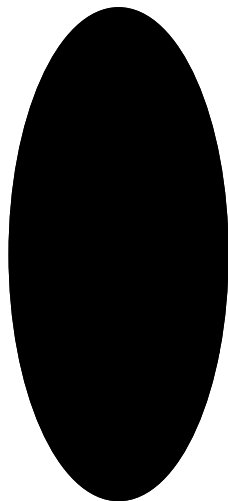
Desired set of logically distinct operators is infinite.

Arbitrary approximation is impossible.

## Theorem 1

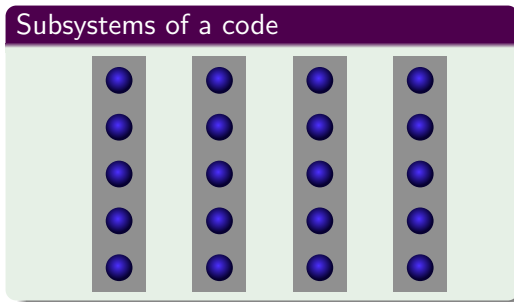
A local-error-detecting code cannot have a universal set of product operators.

Contrasts with a finite **basis** of gates



# The transversal logical operators are not universal

Each transversal part satisfies the detection property.



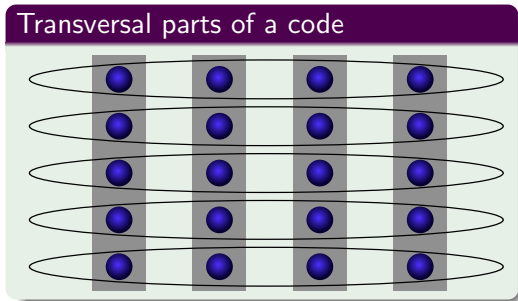
Apply Theorem 1 to the transversal parts.

## Corollary 1

A local-error-detecting code cannot have a universal set of transversal operators.

# The transversal logical operators are not universal

Each transversal part satisfies the detection property.



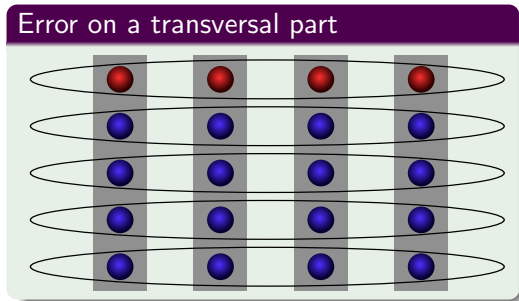
Apply Theorem 1 to the transversal parts.

## Corollary 1

A local-error-detecting code cannot have a universal set of transversal operators.

# The transversal logical operators are not universal

Each transversal part satisfies the detection property.



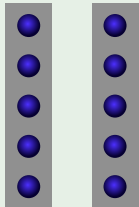
Apply Theorem 1 to the transversal parts.

## Corollary 1

A local-error-detecting code cannot have a universal set of transversal operators.

# Circumventions

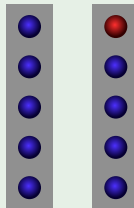
## Non-unitary operators



E.g., measurement and classical feed forward

Requires ancillae

## Stepwise-transversal operators

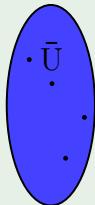


Switch partitions

Error correct in between

Universality?

## Approximate universality

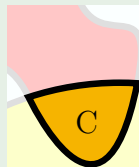


Finite set of logically distinct gates

Non-universal

Accuracy scaling?

## Probabilistic error detection



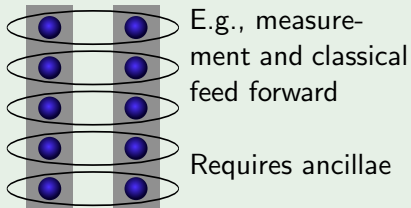
$PEP \propto P$

Transversality & universality?

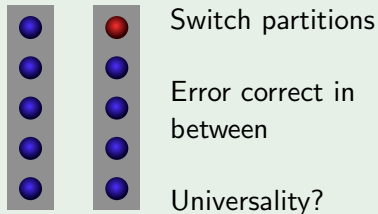
Failure rate scaling?

# Circumventions

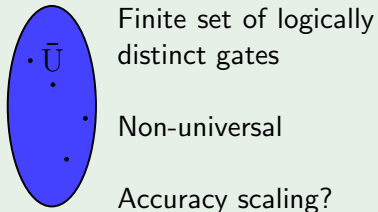
## Non-unitary operators



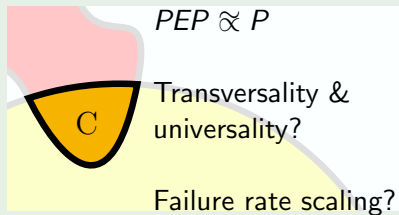
## Stepwise-transversal operators



## Approximate universality

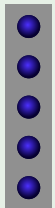


## Probabilistic error detection



# Circumventions

## Non-unitary operators



E.g., measurement and classical feed forward

Requires ancillae

## Stepwise-transversal operators

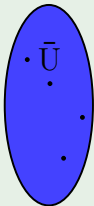


Switch partitions

Error correct in between

Universality?

## Approximate universality

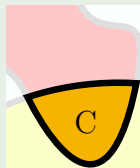


Finite set of logically distinct gates

Non-universal

Accuracy scaling?

## Probabilistic error detection



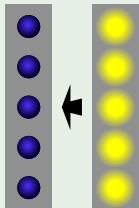
$$PEP \propto P$$

Transversality & universality?

Failure rate scaling?

# Circumventions

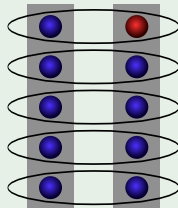
## Non-unitary operators



E.g., measurement and classical feed forward

Requires ancillae

## Stepwise-transversal operators

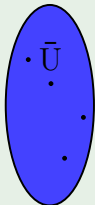


Switch partitions

Error correct in between

Universality?

## Approximate universality

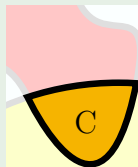


Finite set of logically distinct gates

Non-universal

Accuracy scaling?

## Probabilistic error detection



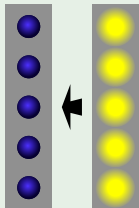
$PEP \propto P$

Transversality & universality?

Failure rate scaling?

# Circumventions

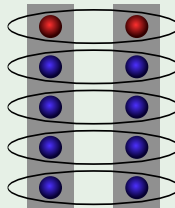
## Non-unitary operators



E.g., measurement and classical feed forward

Requires ancillae

## Stepwise-transversal operators

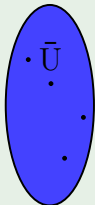


Switch partitions

Error correct in between

Universality?

## Approximate universality



Finite set of logically distinct gates

Non-universal

Accuracy scaling?

## Probabilistic error detection



$$PEP \propto P$$

Transversality & universality?

Failure rate scaling?

# Circumventions

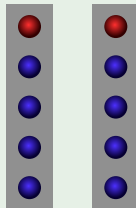
## Non-unitary operators



E.g., measurement and classical feed forward

Requires ancillae

## Stepwise-transversal operators

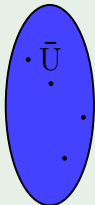


Switch partitions

Error correct in between

Universality?

## Approximate universality

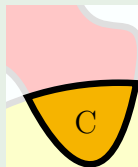


Finite set of logically distinct gates

Non-universal

Accuracy scaling?

## Probabilistic error detection



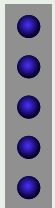
$PEP \propto P$

Transversality & universality?

Failure rate scaling?

# Circumventions

## Non-unitary operators



E.g., measurement and classical feed forward

Requires ancillae

## Stepwise-transversal operators

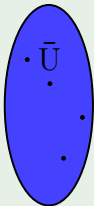


Switch partitions

Error correct in between

Universality?

## Approximate universality

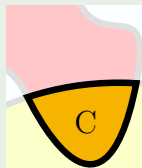


Finite set of logically distinct gates

Non-universal

Accuracy scaling?

## Probabilistic error detection



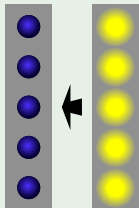
$PEP \propto P$

Transversality & universality?

Failure rate scaling?

# Circumventions

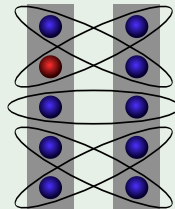
## Non-unitary operators



E.g., measurement and classical feed forward

Requires ancillae

## Stepwise-transversal operators

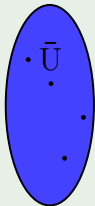


Switch partitions

Error correct in between

Universality?

## Approximate universality

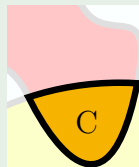


Finite set of logically distinct gates

Non-universal

Accuracy scaling?

## Probabilistic error detection



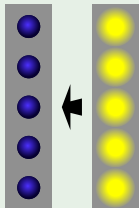
$$PEP \propto P$$

Transversality & universality?

Failure rate scaling?

# Circumventions

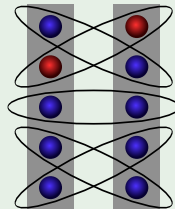
## Non-unitary operators



E.g., measurement and classical feed forward

Requires ancillae

## Stepwise-transversal operators

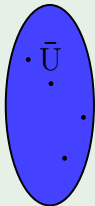


Switch partitions

Error correct in between

Universality?

## Approximate universality

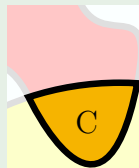


Finite set of logically distinct gates

Non-universal

Accuracy scaling?

## Probabilistic error detection



$$PEP \propto P$$

Transversality & universality?

Failure rate scaling?

# Conclusion

## Result

A local-error-detecting code cannot have a universal set of transversal operators.

## Circumventions

- Non-unitary operators
- Stepwise transversal operators
- Approximate universality
- Probabilistic error detection

## References

- Eastin and Knill, arXiv:0811.4262
- Zeng, Cross, and Chuang, arXiv:0706.1382
- Chen, Chung, Cross, Zeng, and Chuang, arXiv:0801.2360