Classifying Quantum Graph States

Nathan Claudet

29/09/26
CANA seminar - Marseille



Quantum computing

Computing with quantum physics

- Quantum physics = very small scale.

Computing with quantum physics

- Quantum physics = very small scale.
- The laws of quantum physics differ from laws of classical physics (no determinism).

Computing with quantum physics

- Quantum physics = very small scale.
- The laws of quantum physics differ from laws of classical physics (no determinism).
- Quantum computing = using quantum physical phenomena to encode and manipulate information.

Computing with quantum physics

- Quantum physics = very small scale.
- The laws of quantum physics differ from laws of classical physics (no determinism).
- Quantum computing = using quantum physical phenomena to encode and manipulate information.

Theorem (Shor's algorithm, 1994)

There exist an **efficient** *quantum algorithm that finds the prime factors of an integer.*

$$15 = 5 \times 3$$

$$221 = 13 \times 17$$

$$269535011 = 12923 \times 20857$$

How to build a quantum computer

- In classical computing, bits: 0 or 1.
- In quantum computing, quantum bits (qubits): $|0\rangle$, $|1\rangle$, but also $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$.

How to build a quantum computer

- In classical computing, bits: 0 or 1.
- In quantum computing, quantum bits (qubits): $|0\rangle$, $|1\rangle$, but also $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$.

Many possible implementation of a qubit:

- the energy of an ion;
- the spin of an electron;
- the polarisation of a photon.

How to build a quantum computer

- In classical computing, bits: 0 or 1.
- In quantum computing, quantum bits (qubits): $|0\rangle$, $|1\rangle$, but also $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$.

Many possible implementation of a qubit:

- the energy of an ion;
- the spin of an electron;
- the polarisation of a photon.

+ ways to interact with the qubits to create **entanglement**.

Quantum entanglement

- Quantum states are composed of qubits.





$|0\rangle \otimes |0\rangle = |00\rangle$

Quantum entanglement

- Quantum states are composed of qubits.
- Unitaries: transformations between quantum states.





$$|0\rangle \otimes |0\rangle = |00\rangle$$

Quantum entanglement

- Quantum states are composed of qubits.
- Unitaries: transformations between quantum states.
- Example 1: the Hadamard gate H. $\rightarrow$ H $|0\rangle = |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$

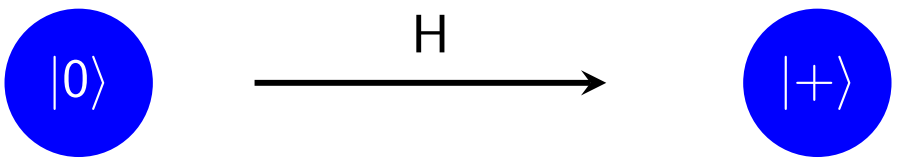


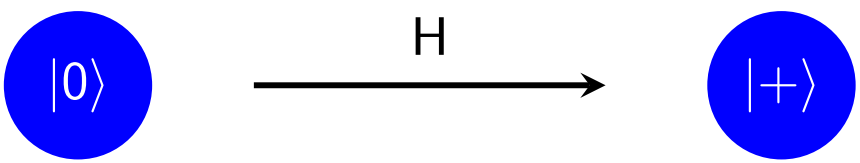


$|0\rangle \otimes |0\rangle = |00\rangle$

Quantum entanglement

- Quantum states are composed of qubits.
- Unitaries: transformations between quantum states.
- Example 1: the Hadamard gate H. $\rightarrow$ H $|0\rangle = |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$



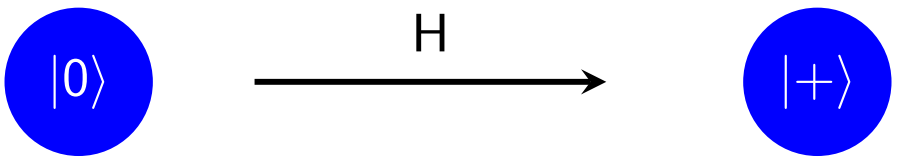


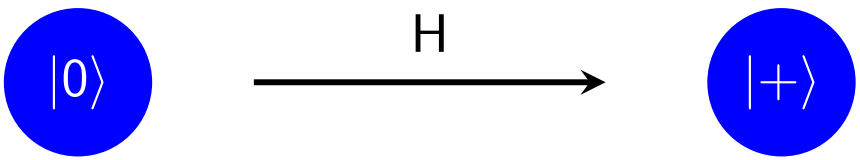
$|0\rangle \otimes |0\rangle = |00\rangle$

$$|+\rangle \otimes |+\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$$

Quantum entanglement

- Quantum states are composed of qubits.
- Unitaries: transformations between quantum states.
- Example 1: the Hadamard gate H. $\rightarrow$ H $|0\rangle = |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$
- Example 2: the controled-Z gate CZ. $\rightarrow$ CZ $|11\rangle = -|11\rangle$



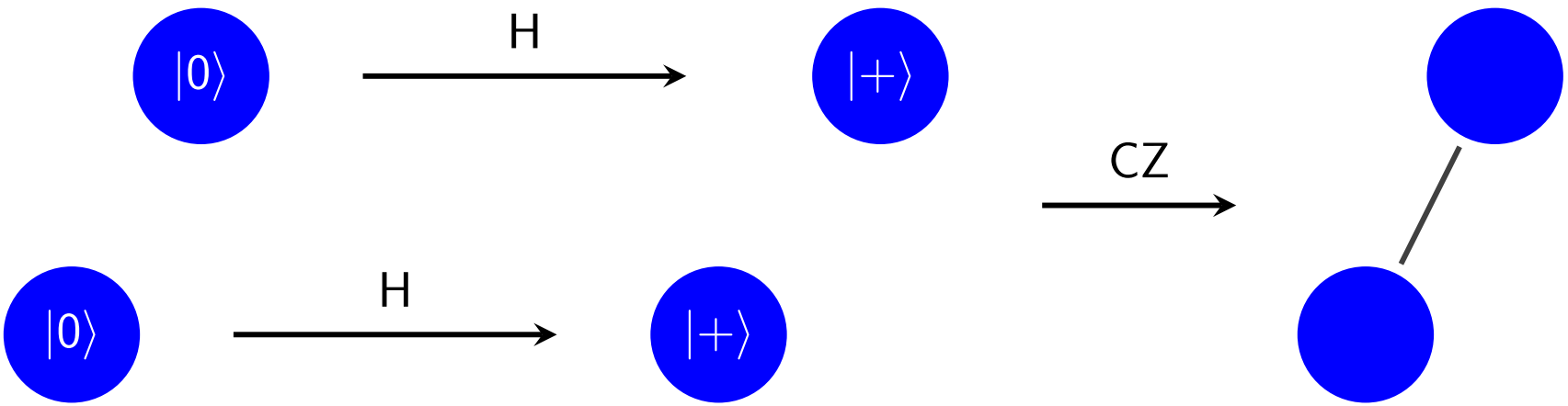


$|0\rangle \otimes |0\rangle = |00\rangle$

$$|+\rangle \otimes |+\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$$

Quantum entanglement

- Quantum states are composed of qubits.
- Unitaries: transformations between quantum states.
- Example 1: the Hadamard gate H. $\rightarrow$ H $|0\rangle = |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$
- Example 2: the controled-Z gate CZ. $\rightarrow$ CZ $|11\rangle = -|11\rangle$



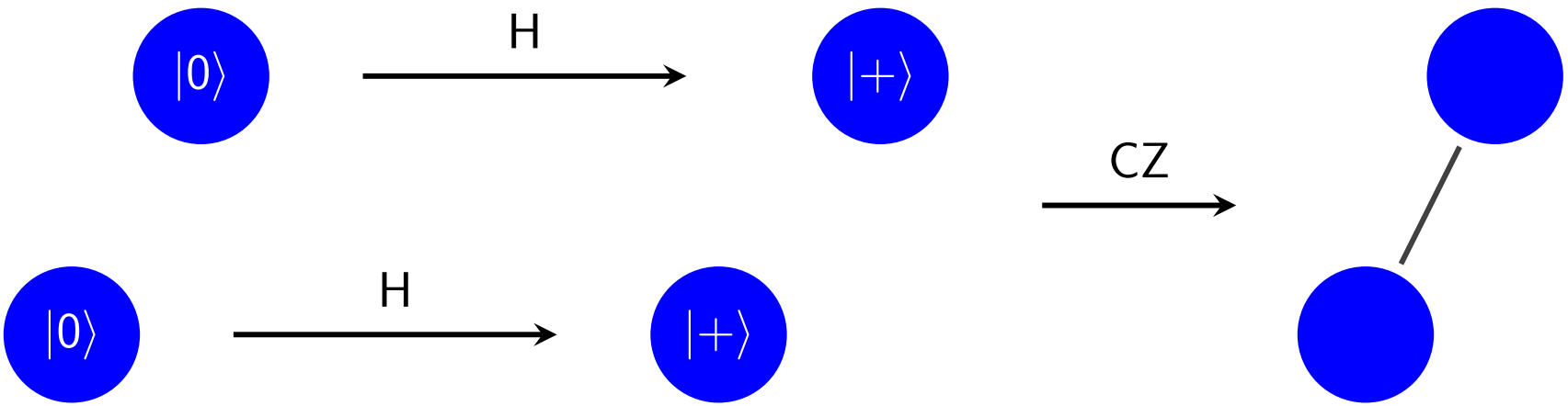
$|0\rangle \otimes |0\rangle = |00\rangle$

$|+\rangle \otimes |+\rangle$
$= \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$

$\frac{1}{2}(|00\rangle + |01\rangle + |10\rangle - |11\rangle)$

Quantum entanglement

- Quantum states are composed of qubits.
- Unitaries: transformations between quantum states.
- Example 1: the Hadamard gate H. $\rightarrow$ H $|0\rangle = |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$
- Example 2: the controled-Z gate CZ. $\rightarrow$ CZ $|11\rangle = -|11\rangle$



$|0\rangle \otimes |0\rangle = |00\rangle$

not entangled

$|+\rangle \otimes |+\rangle$
$= \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$

not entangled

$\frac{1}{2}(|00\rangle + |01\rangle + |10\rangle - |11\rangle)$

entangled

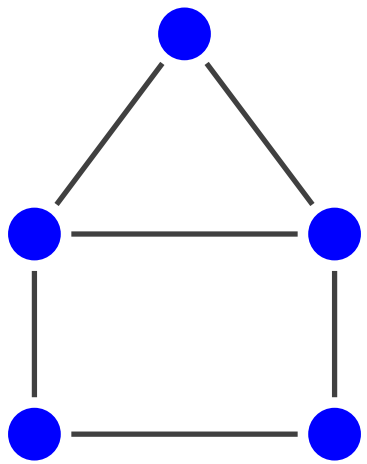
Qubits are **entangled** when they cannot be described as separate entities.

Graph states

Graphs

Definition

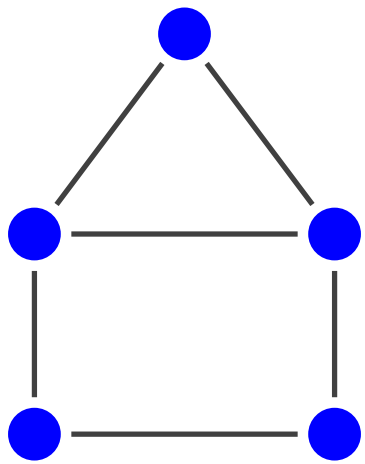
A graph is composed of two sets, a set of vertices and a set of edges linking vertices.



Graphs

Definition

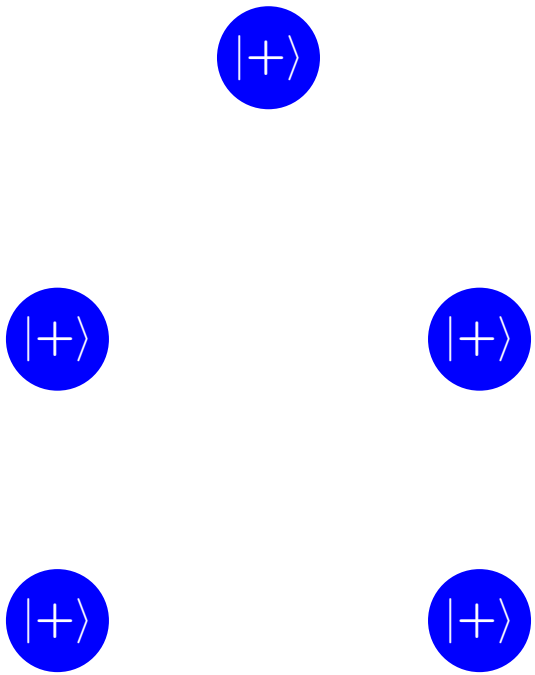
A graph is composed of two sets, a set of vertices and a set of edges linking vertices.



- We consider graphs that are simple (no multiples edges, no loops) and undirected (edges do not have a direction).

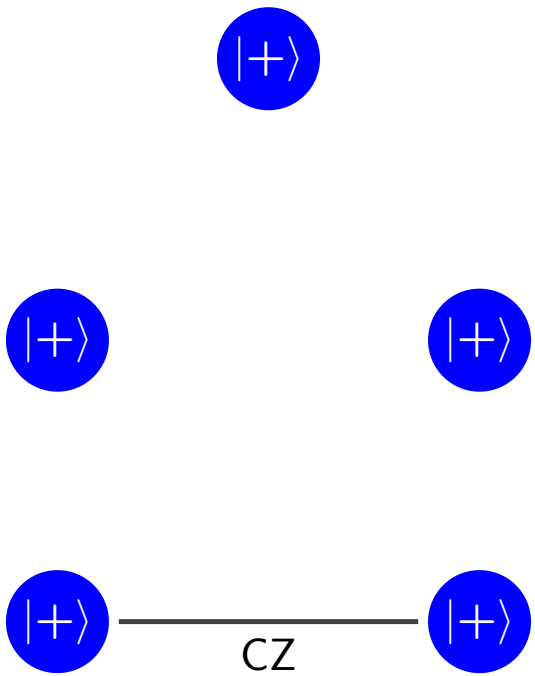
Graph states

A graph state is a quantum state represented by a graph. The vertices represent the qubits and the edges represent entanglement.



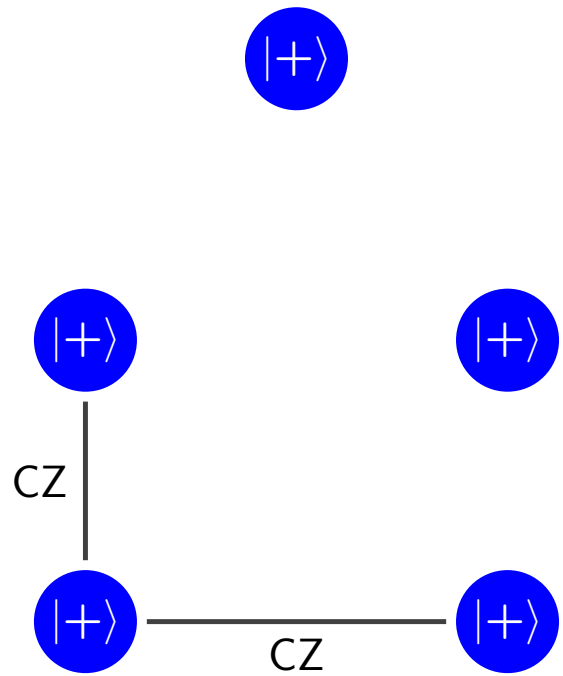
Graph states

A graph state is a quantum state represented by a graph. The vertices represent the qubits and the edges represent entanglement.



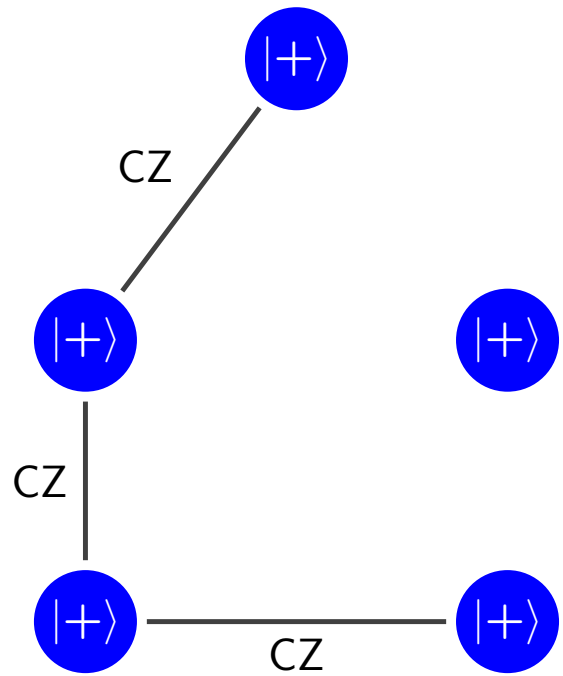
Graph states

A graph state is a quantum state represented by a graph. The vertices represent the qubits and the edges represent entanglement.



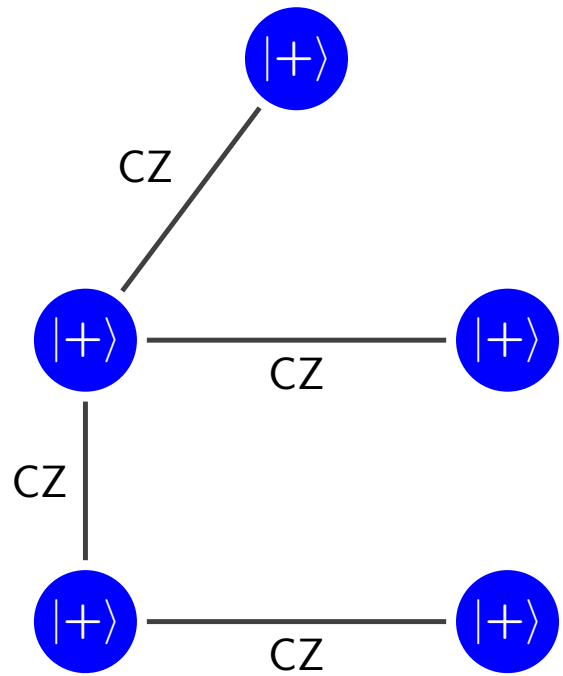
Graph states

A graph state is a quantum state represented by a graph. The vertices represent the qubits and the edges represent entanglement.



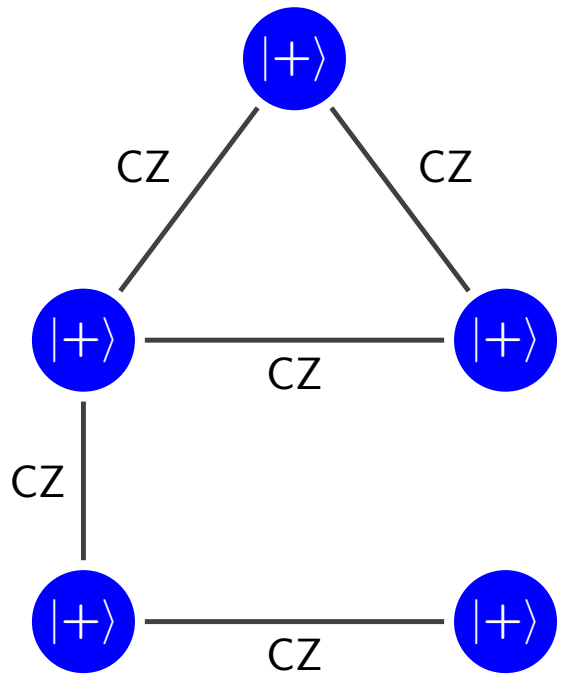
Graph states

A graph state is a quantum state represented by a graph. The vertices represent the qubits and the edges represent entanglement.



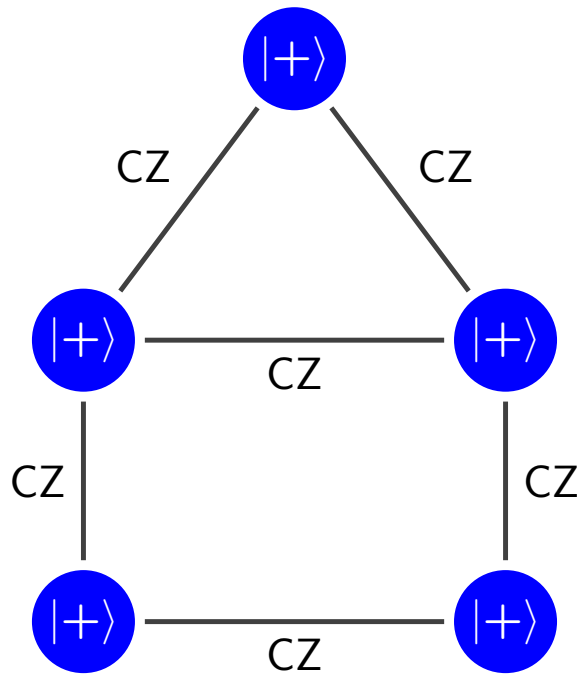
Graph states

A graph state is a quantum state represented by a graph. The vertices represent the qubits and the edges represent entanglement.



Graph states

A graph state is a quantum state represented by a graph. The vertices represent the qubits and the edges represent entanglement.



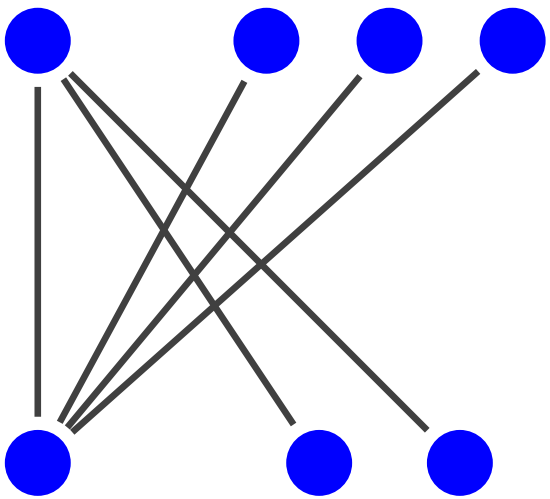
Bijection

Proposition

Graph states are in one-to-one correspondance with graphs.

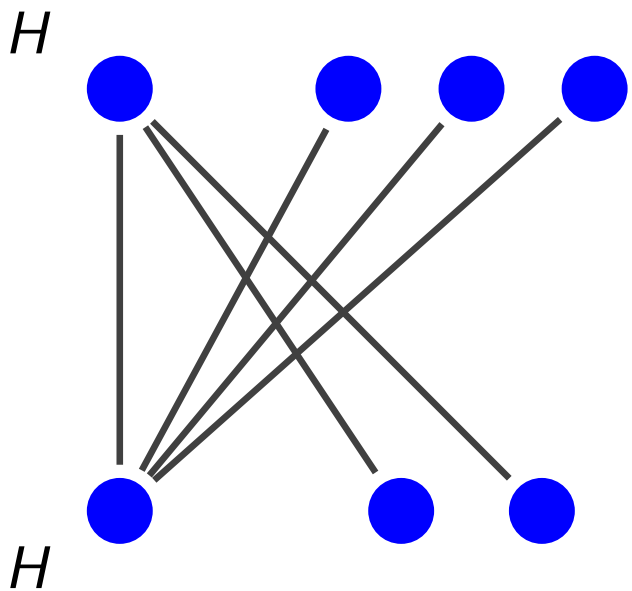
Single-qubit operations

Single-qubit unitary gates are considered “free”.



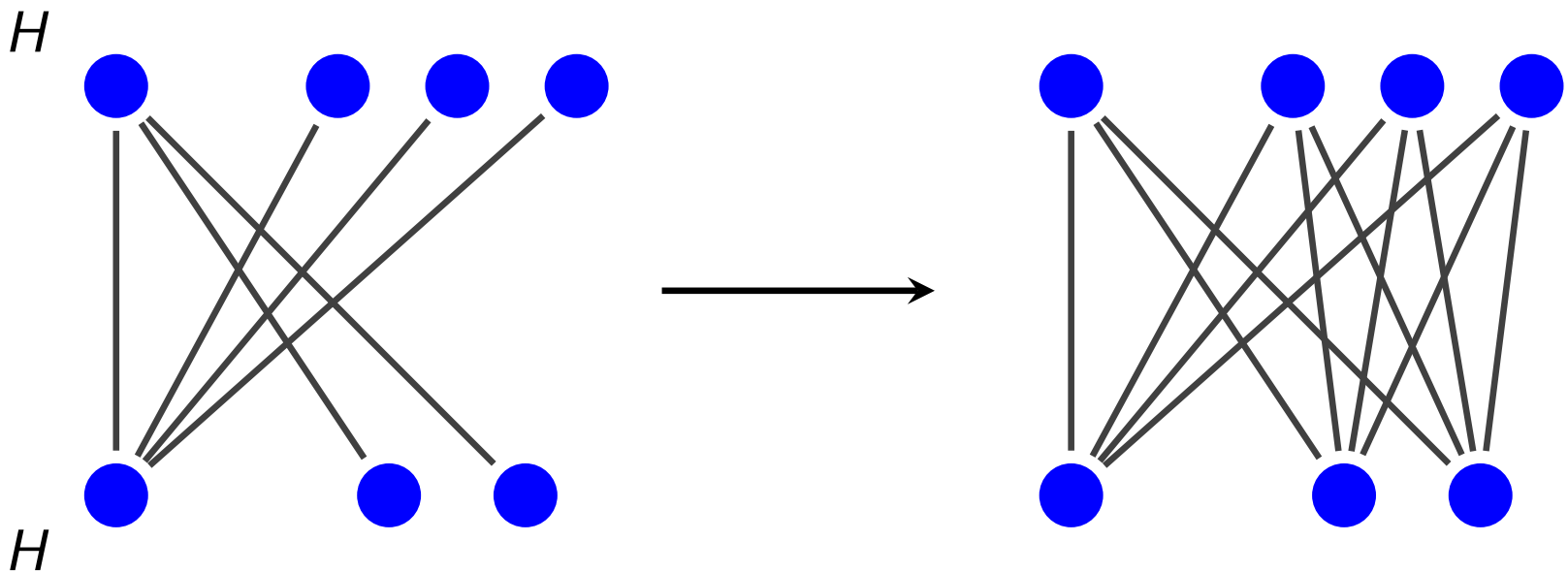
Single-qubit operations

Single-qubit unitary gates are considered "free".



Single-qubit operations

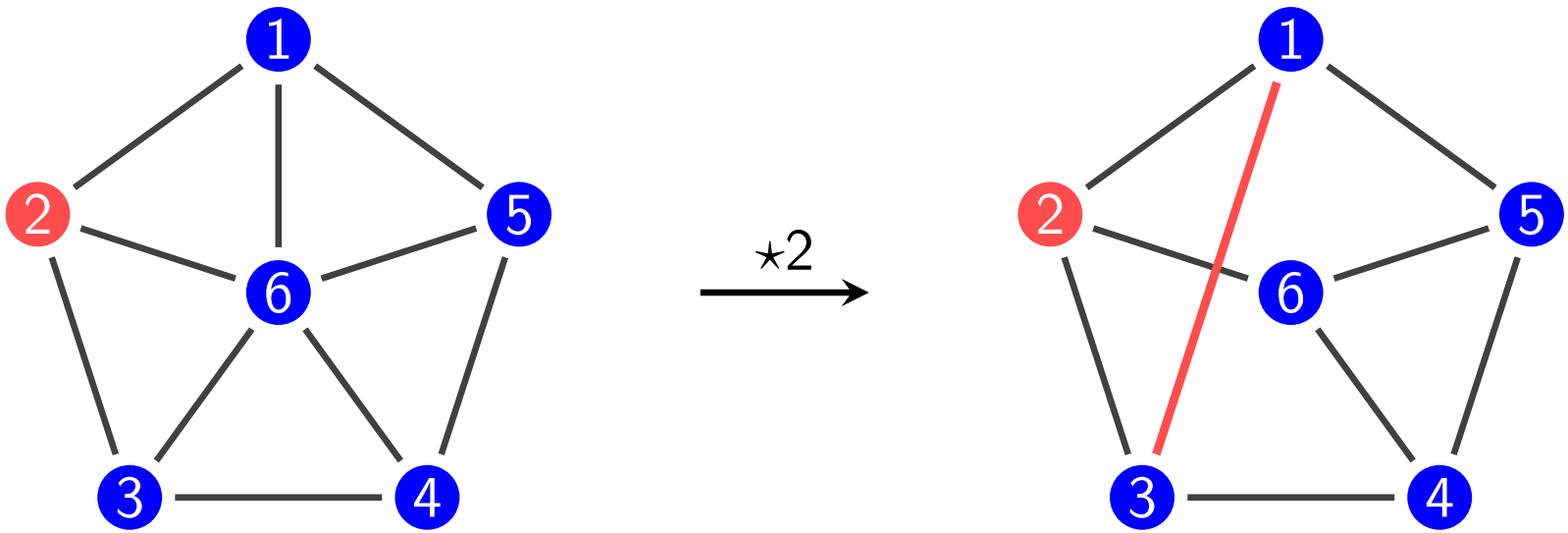
Single-qubit unitary gates are considered "free".



Local complementation

Definition (Kotzig, 1966)

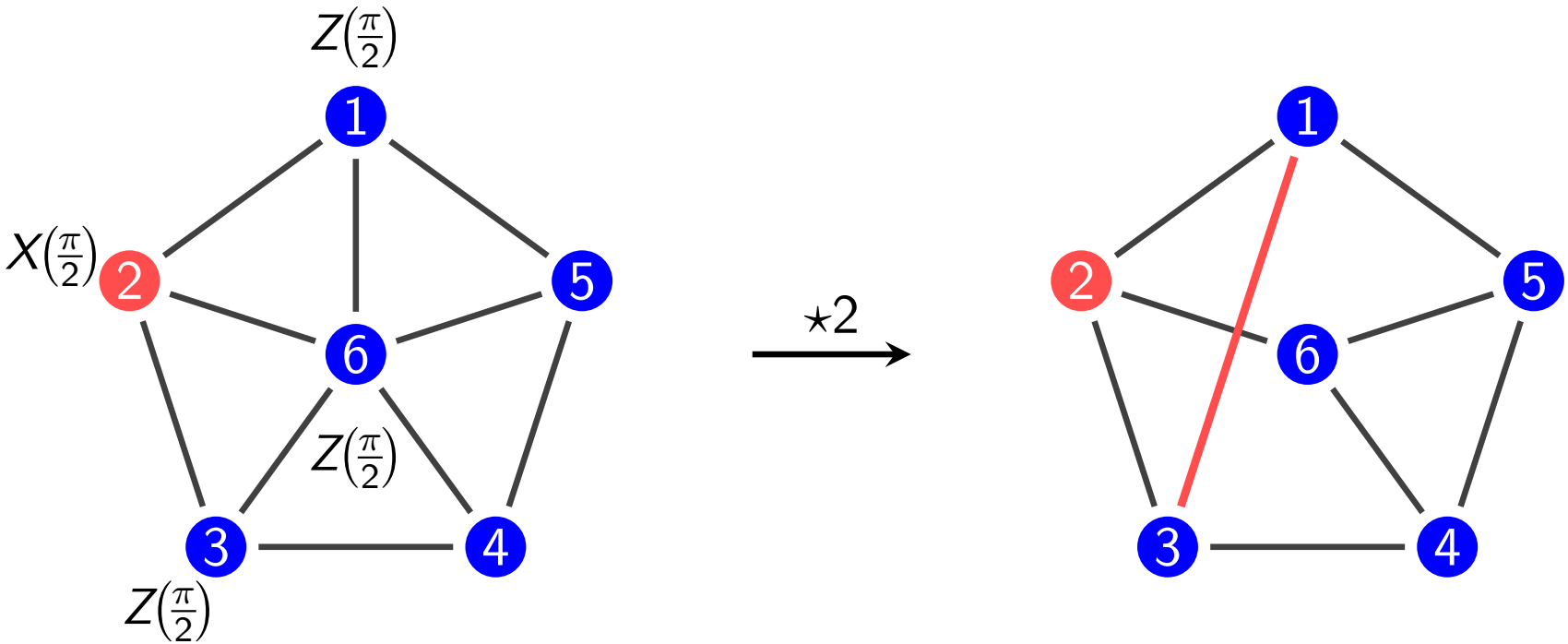
A local complementation on a vertex u consists in complementing the (open) neighbourhood of u.



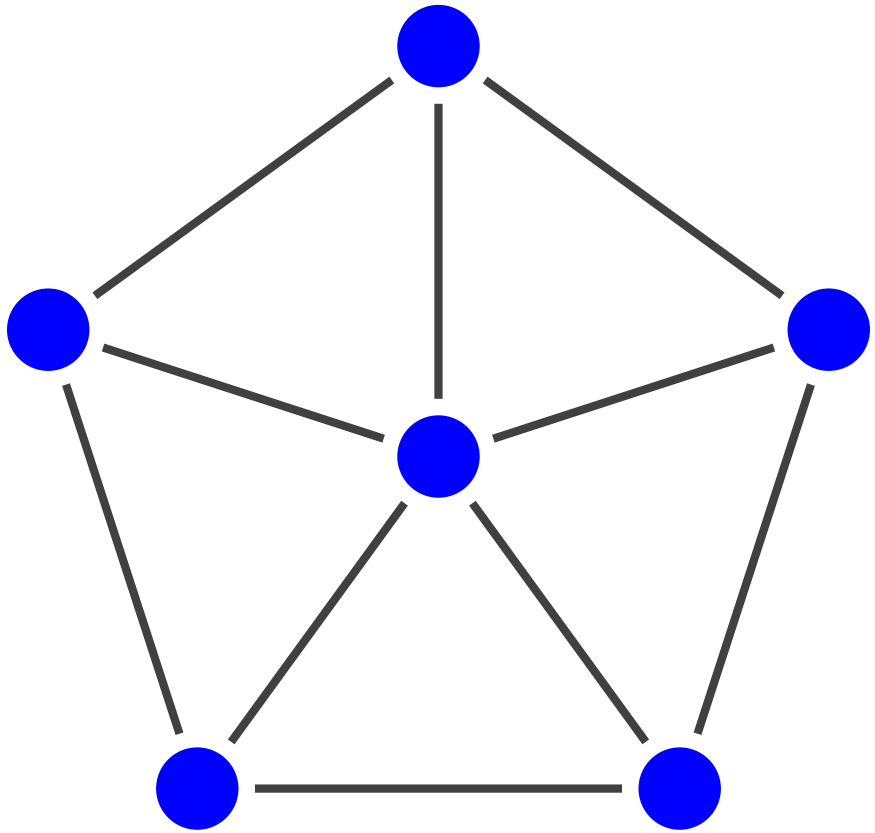
Local complementation

Definition (Kotzig, 1966)

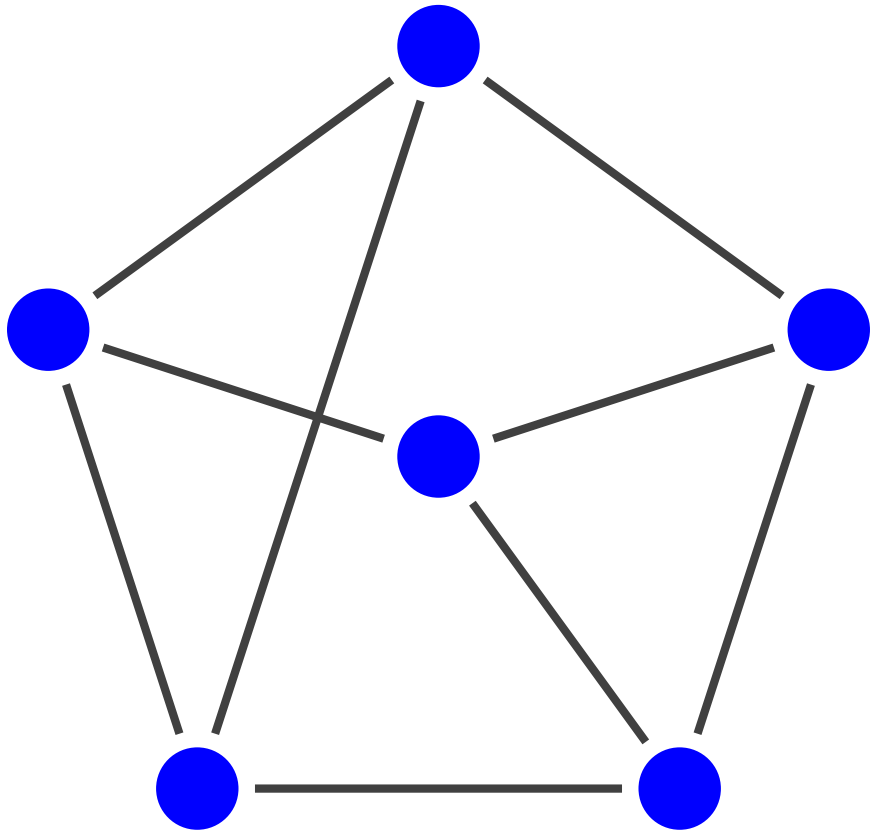
A local complementation on a vertex u consists in complementing the (open) neighbourhood of u.



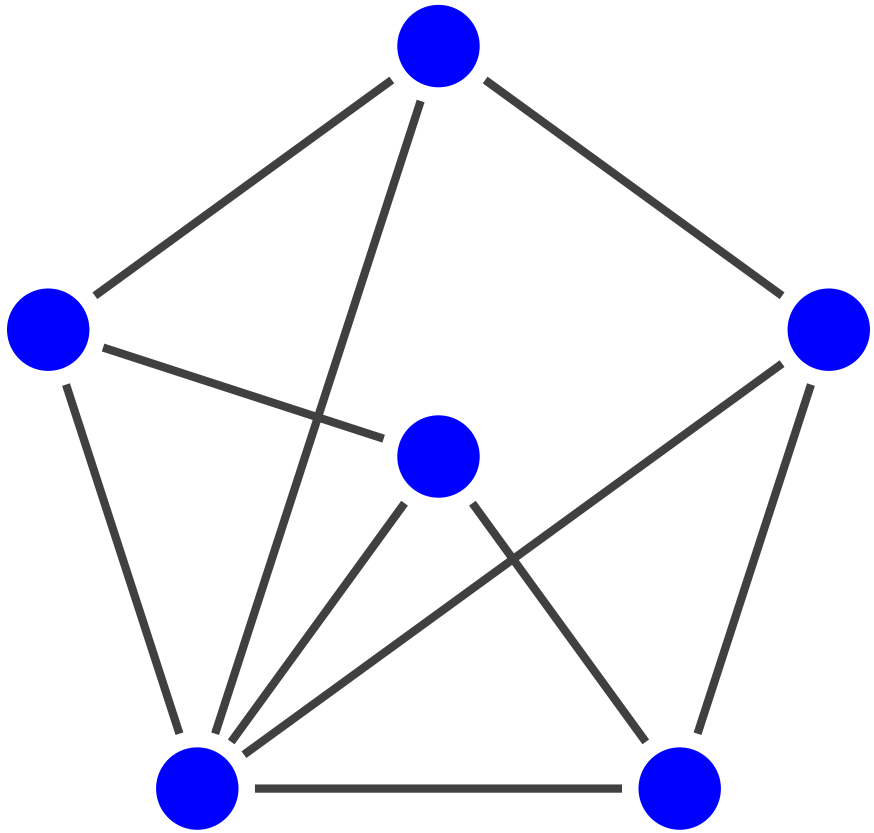
Finding equivalent graph states



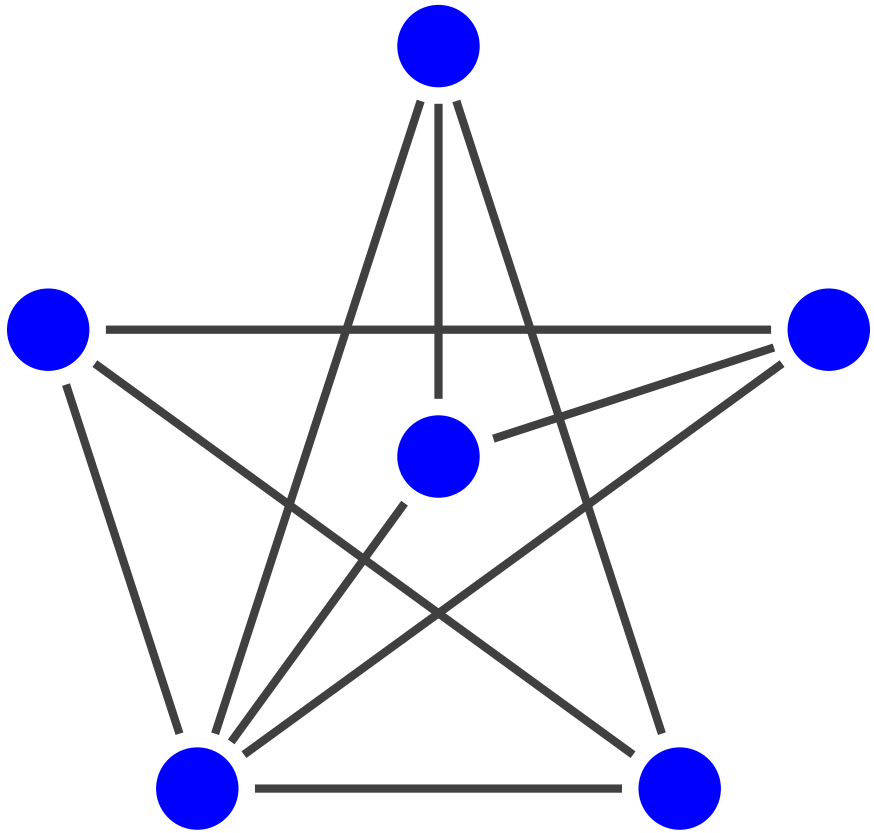
Finding equivalent graph states



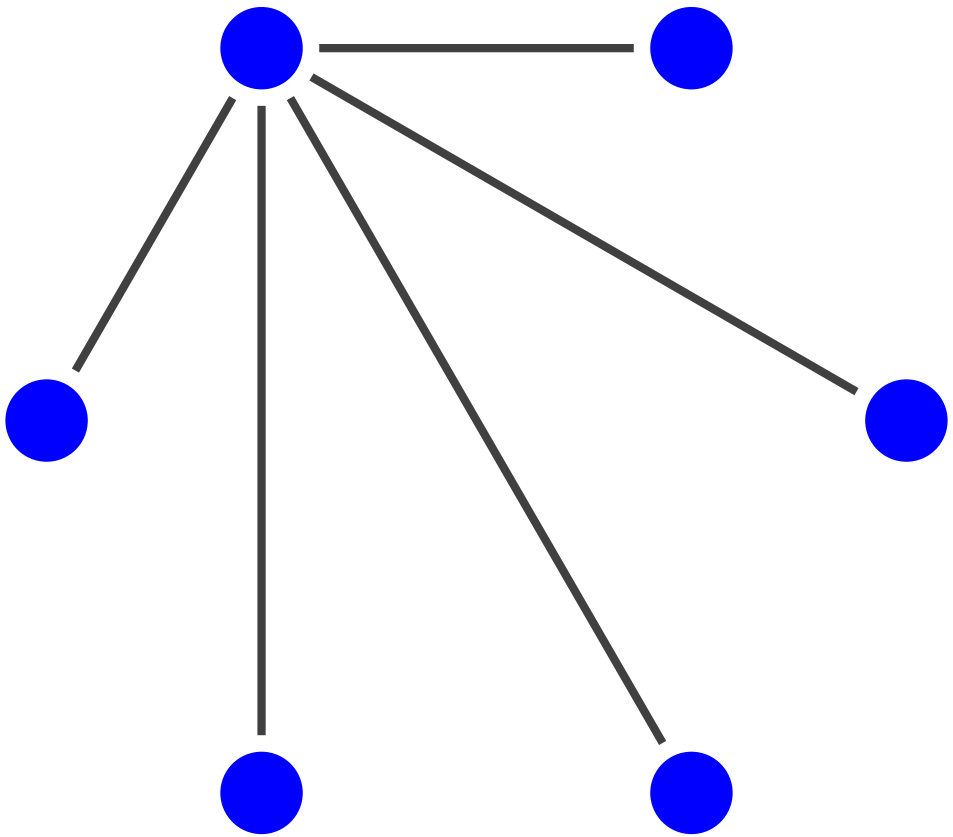
Finding equivalent graph states



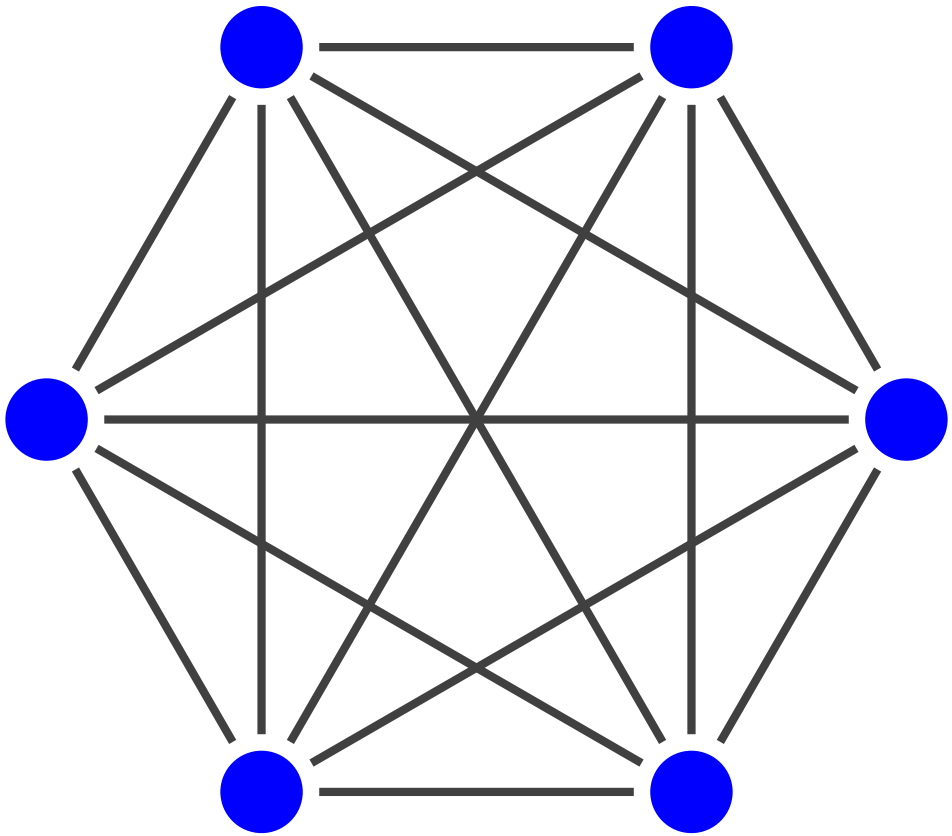
Finding equivalent graph states



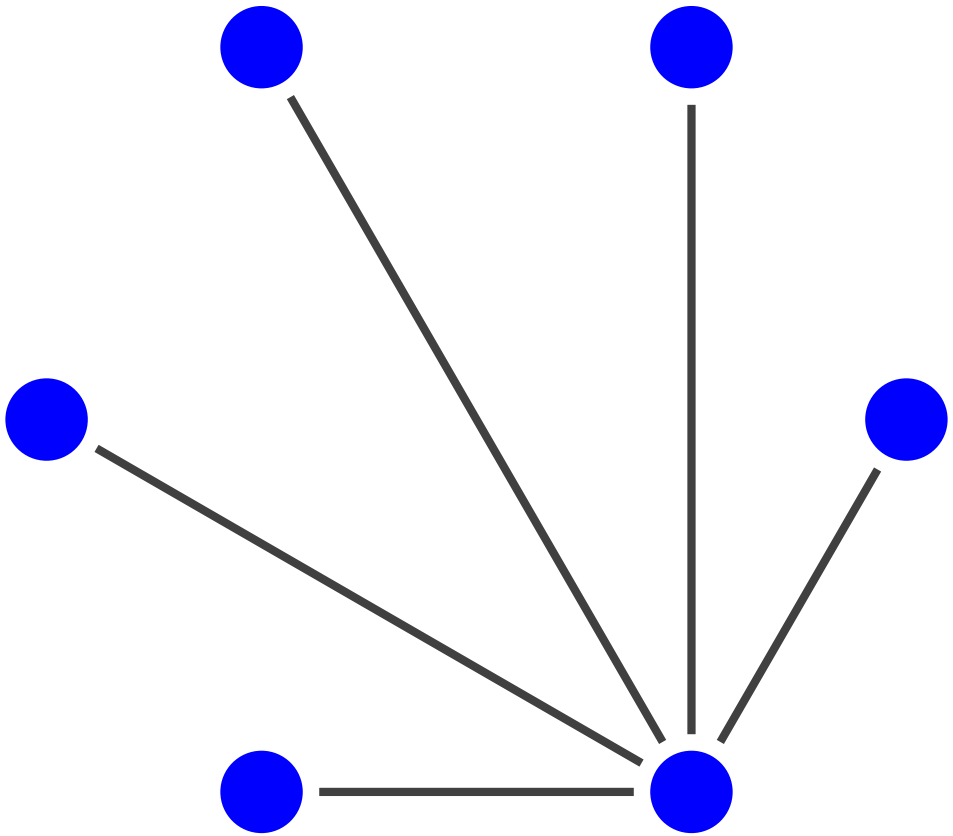
Finding equivalent graph states



Finding equivalent graph states



Finding equivalent graph states



Structure of the presentation

Structure of the presentation

Classifying graph states...

- ...according to their computational power as a quantum resource.

- ...according to their entanglement.

Structure of the presentation

Classifying graph states...

- ...according to their computational power as a quantum resource. **Which graph states are good resources for measurement-based quantum computation?**

- ...according to their entanglement.

Structure of the presentation

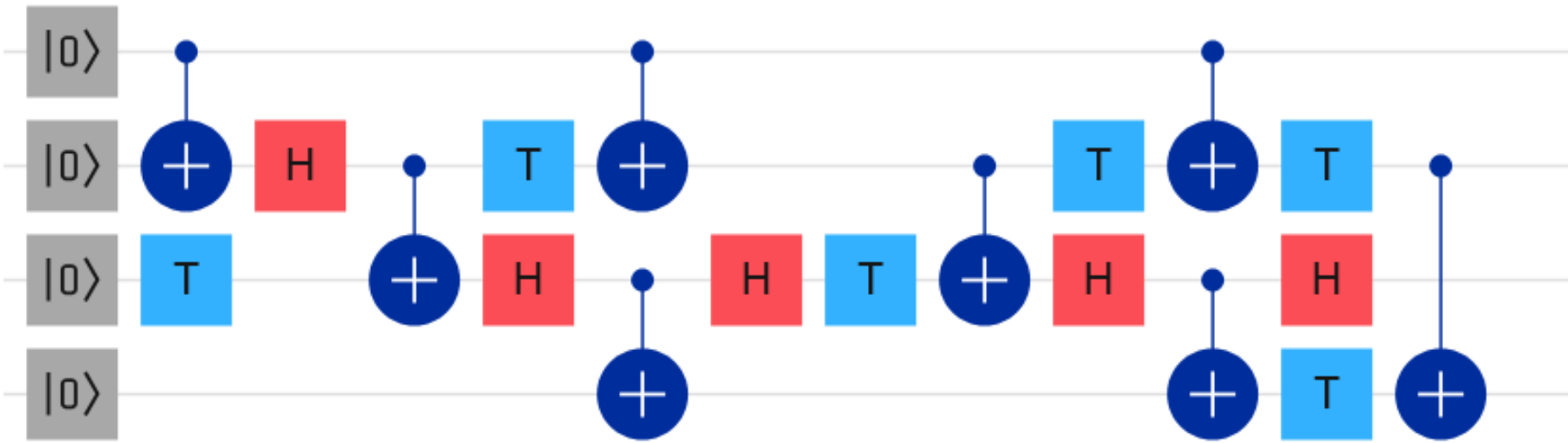
Classifying graph states...

- ...according to their computational power as a quantum resource.
  Which graph states are good resources for measurement-based quantum computation?

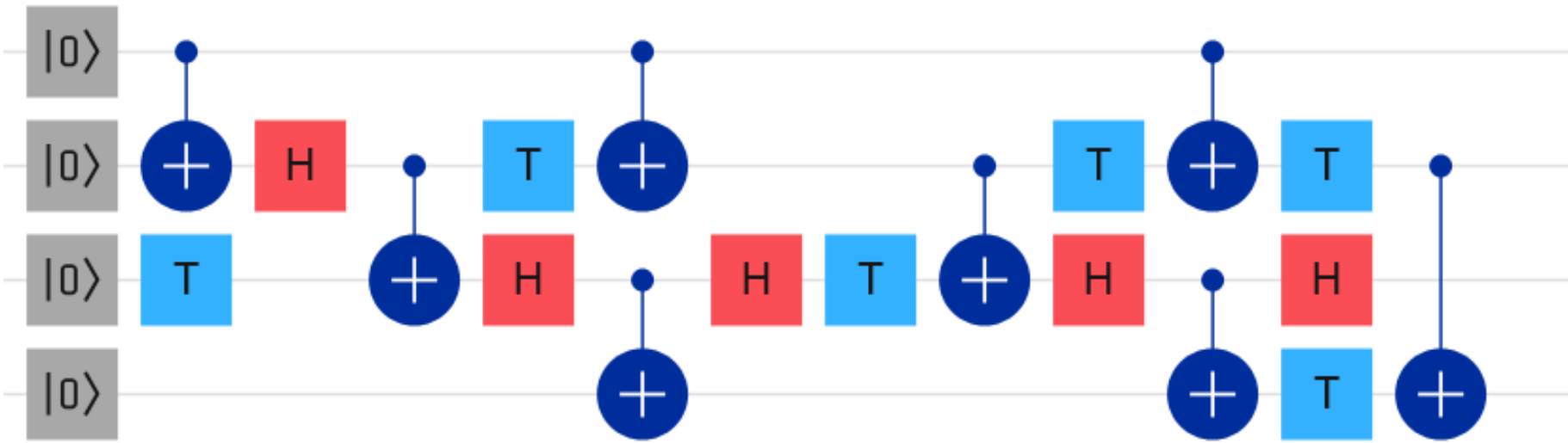
- ...according to their entanglement.
  When are two graph states equivalent?

Classifying graph states according to their computational power as a quantum resource

The quantum circuit model

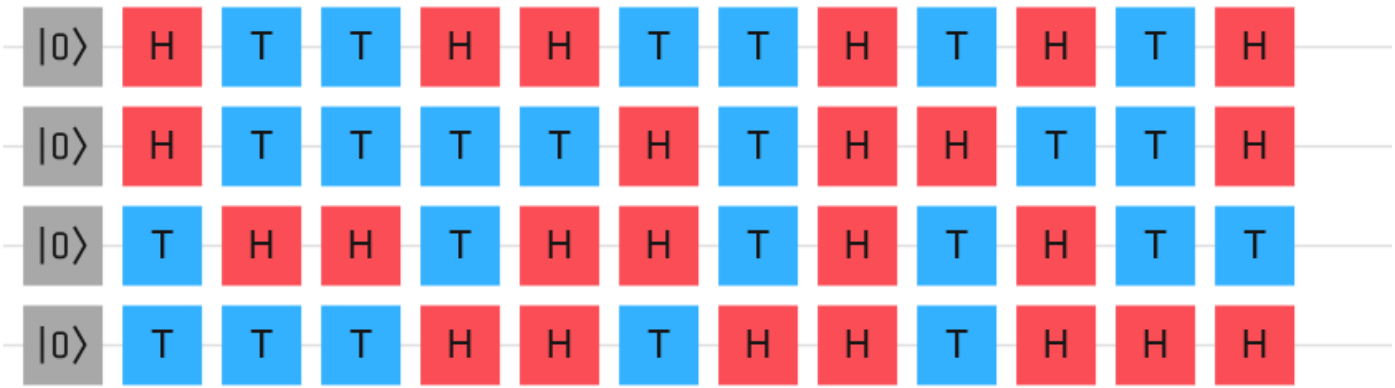


The quantum circuit model



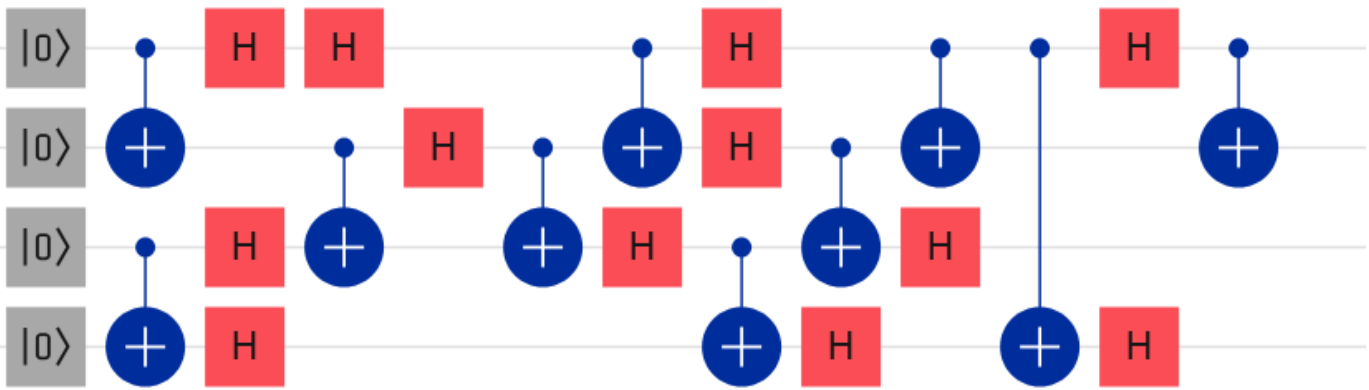
What features make quantum computers (allegedly) more powerful than classical computers?

Quantum circuits with only local gates



No entanglement $\rightarrow$ The quantum computation can be efficiently simulated on a classical computer.

Quantum circuits with only Clifford gates



Theorem (Gottesman-Knill, 1998)

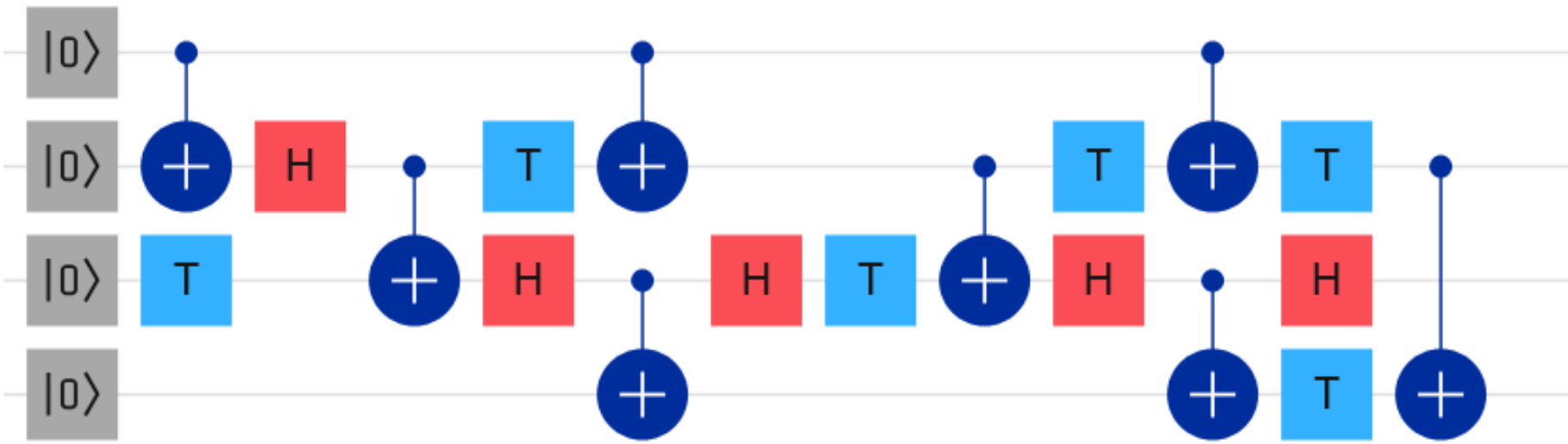
A quantum computation consisting only of Clifford gates can be efficiently simulated on a classical computer.

Sources of computational power

Two potential sources of quantum computational power:

- 1) Entanglement;
- 2) non-Clifford operations.

The quantum circuit model



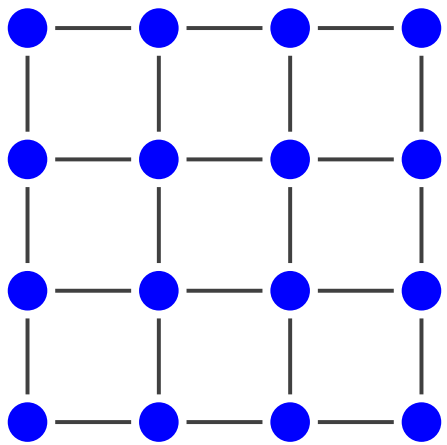
Measurement-based quantum computing

The idea behind **measurement-based quantum computing** is to separate the two potential sources of quantum computational power.

Measurement-based quantum computing

The idea behind **measurement-based quantum computing** is to separate the two potential sources of quantum computational power.

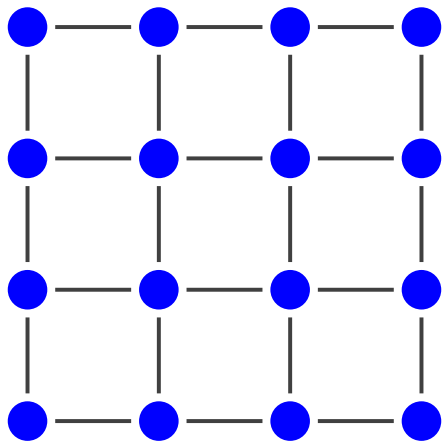
- 1) Prepare a **graph state** with only CZ gates (Clifford gates);



Measurement-based quantum computing

The idea behind **measurement-based quantum computing** is to separate the two potential sources of quantum computational power.

- 1) Prepare a **graph state** with only CZ gates (Clifford gates);
- 2) **Measure** qubits in some ways that may depend on the previous measurements (no entanglement added).



Reformulation of the question

In measurement-based quantum computation, the question

"**What features make quantum computers powerful?**"

becomes

"**Which graph states are universal resources?**"

Reformulation of the question

In measurement-based quantum computation, the question

"**What features make quantum computers powerful?**"

becomes

"**Which graph states are universal resources?**"

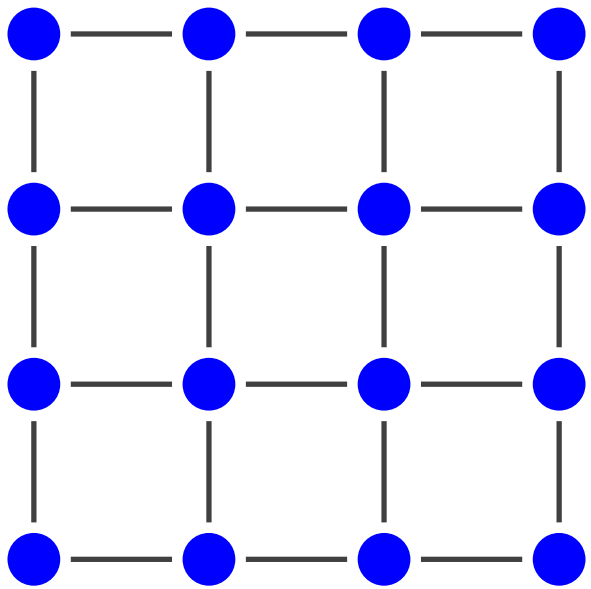
Alternatively:

"**For which graph states is measurement-based quantum computation efficiently classically simulable?**"

Power of measurement-based quantum computing

Theorem (Briegel, Raussendorf, 2001)

Measurement-based quantum computing on the grid graph states has the same computational power as quantum circuits. **(Good resource)**



Power of measurement-based quantum computing

Theorem (Nielsen, 2005)

For path graph states, measurement-based quantum computing is efficiently simulable on a classical computer. **(Bad resource)**



Power of measurement-based quantum computing

Theorem (Nielsen, 2005)

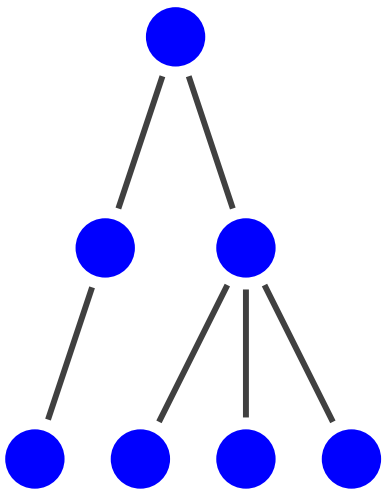
For path graph states, measurement-based quantum computing is efficiently simulable on a classical computer. **(Bad resource)**



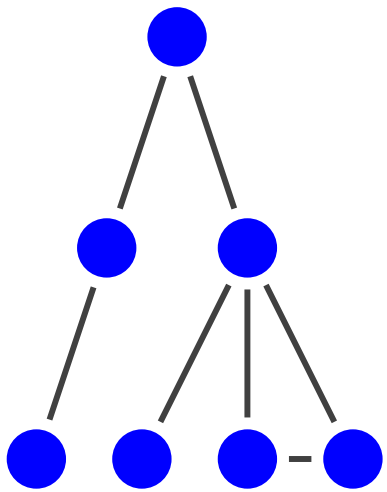
Intuition: no quantum power when the graph looks like a tree.

Tree-width

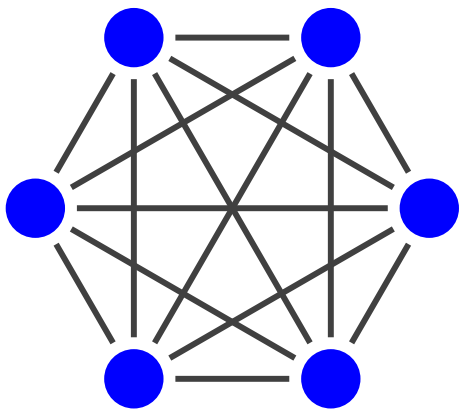
Informally, tree-width is a function in $\mathbb{N}$ that measures how close a graph is to a tree.



tree-width $= 1$



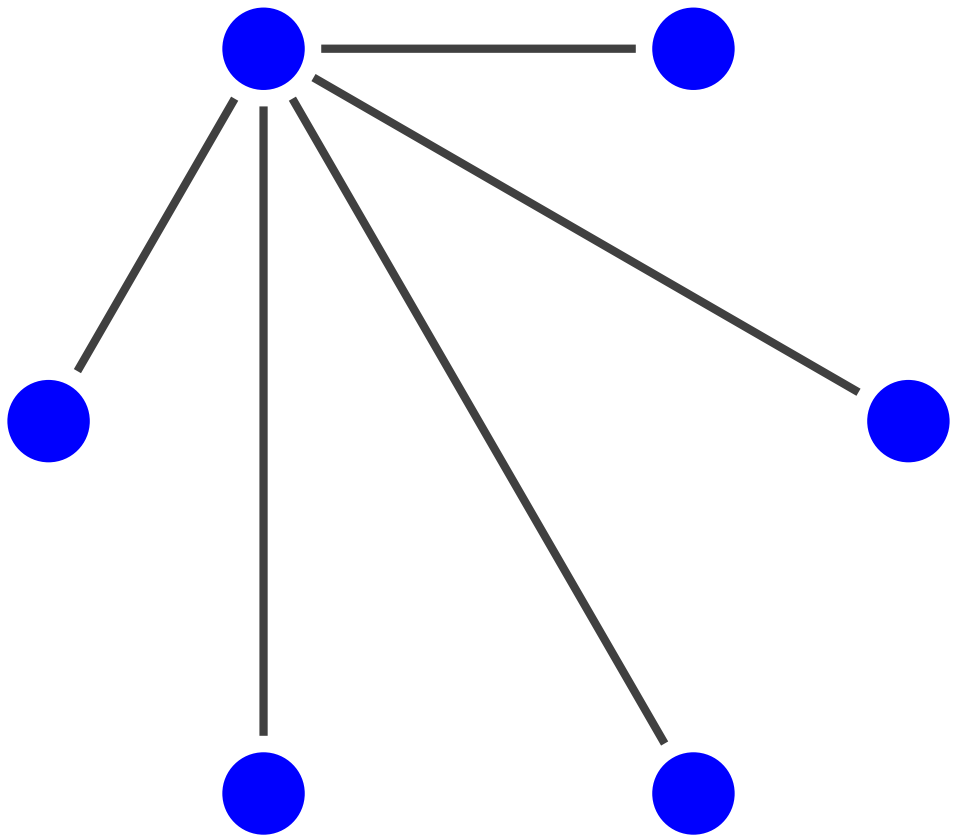
tree-width $= 2$



complete graph
on n vertices:
tree-width $= n - 1$

Tree-width does not pair well with local complementation

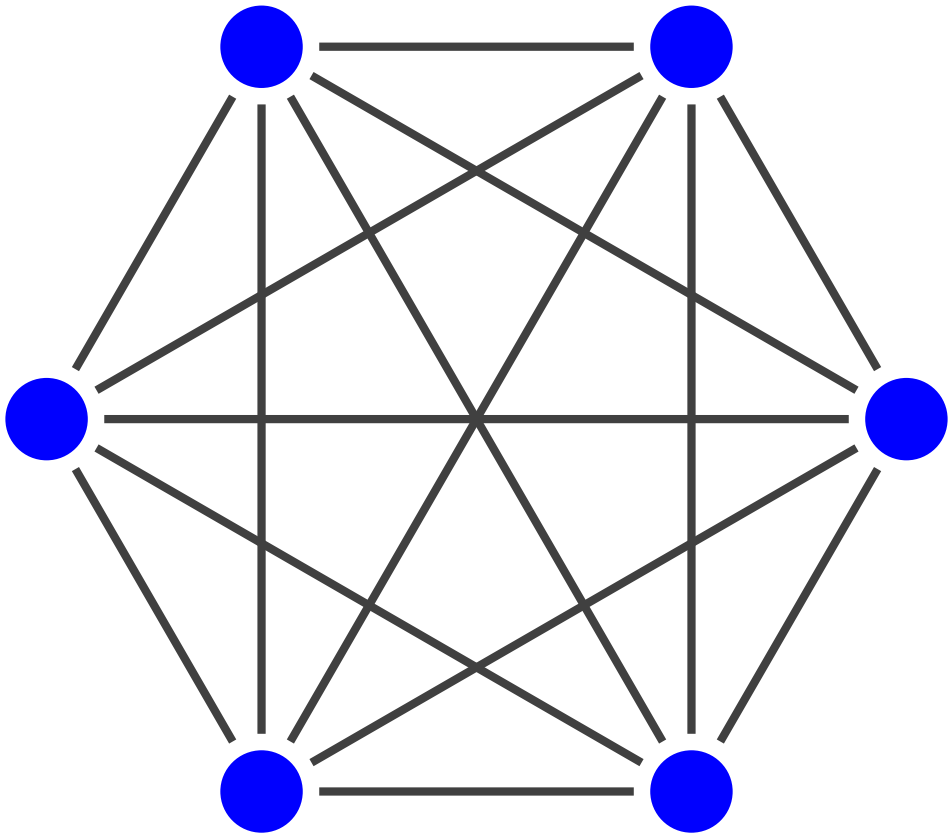
A star and a complete graph are the same up to local complementation.



tree-width $= 1$

Tree-width does not pair well with local complementation

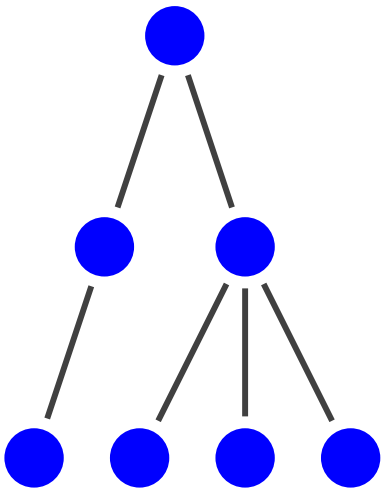
A star and a complete graph are the same up to local complementation.



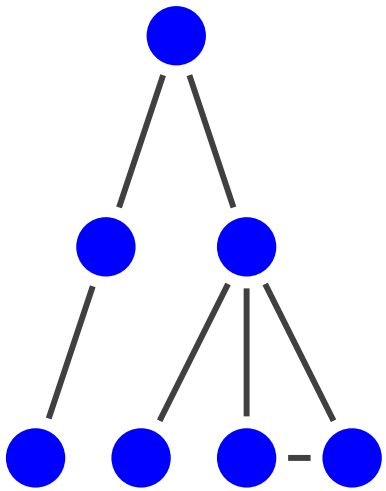
tree-width $= n - 1$

Rank-width

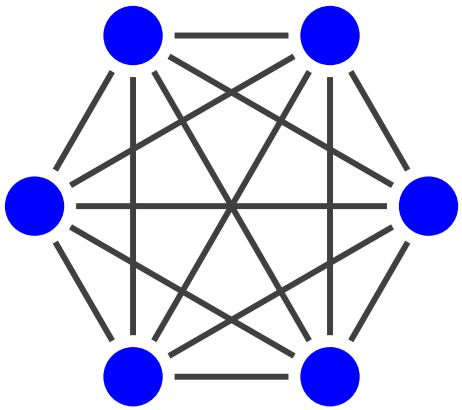
Informally, rank-width is like tree-width, but invariant by local complementation.



rank-width $= 1$



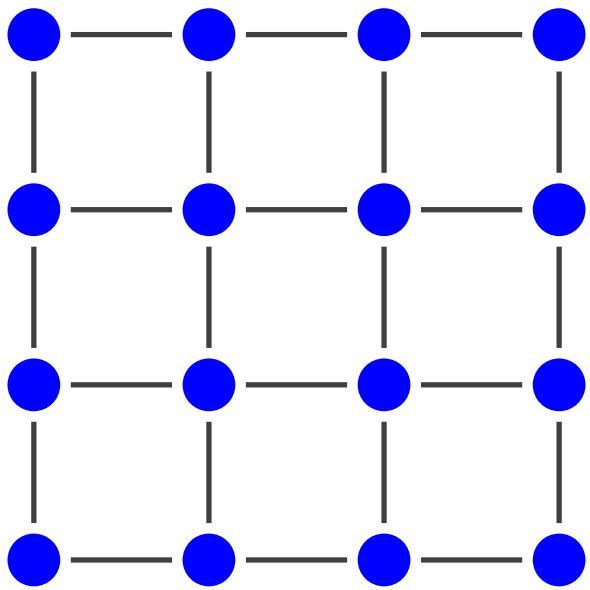
rank-width $= 2$



complete graph on n vertices: rank-width $= 1$

Rank-width of the square grid

The $m \times m$ square grid has rank-width $m - 1$ (Jelínek, 2008).



Low rank-width implies efficient classical simulation

Theorem (Van den Nest et al., 2006)

Measurement-based quantum computing on graph states with **bounded rank-width** *is efficiently simulable on a classical computer.*

Small rank-width = efficient classical simulation?

Small rank width implies efficient classical simulation.

Small rank-width = efficient classical simulation?

Small rank width implies efficient classical simulation.

Is it an equivalence?

Small rank-width = efficient classical simulation?

Small rank width implies efficient classical simulation.

Is it an equivalence?

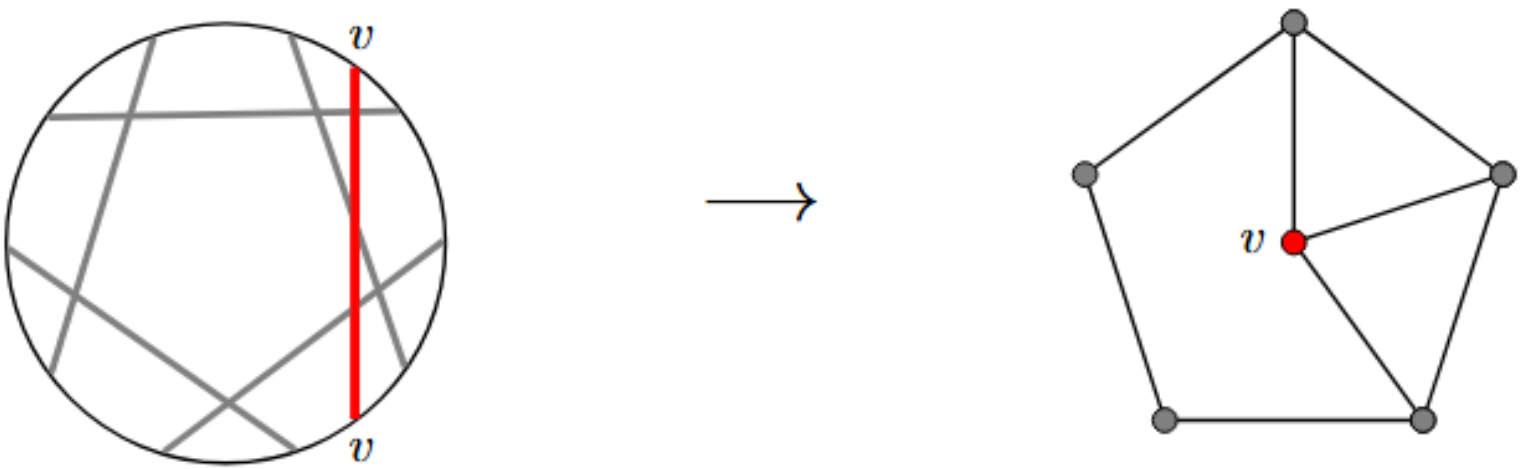
No!

Circle graphs

Circle graphs

Definition

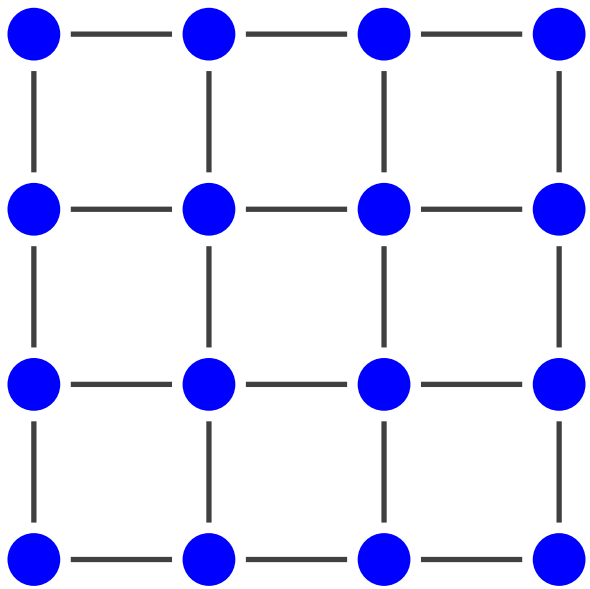
A circle graph is the intersection graph of a chord diagram.



Figures: Rose McCarty, PhD Thesis

Not all graphs are circle graphs

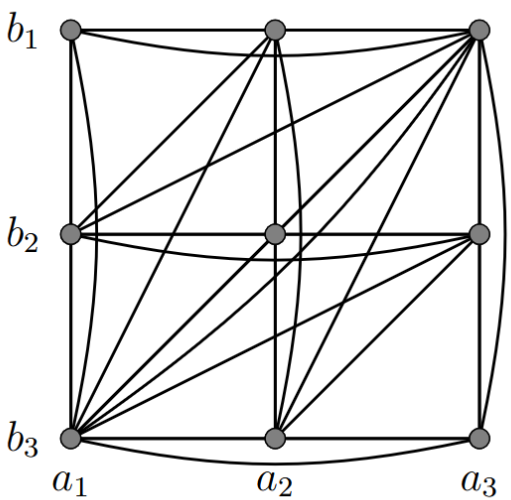
The square grid cannot be represented as a chord diagram.

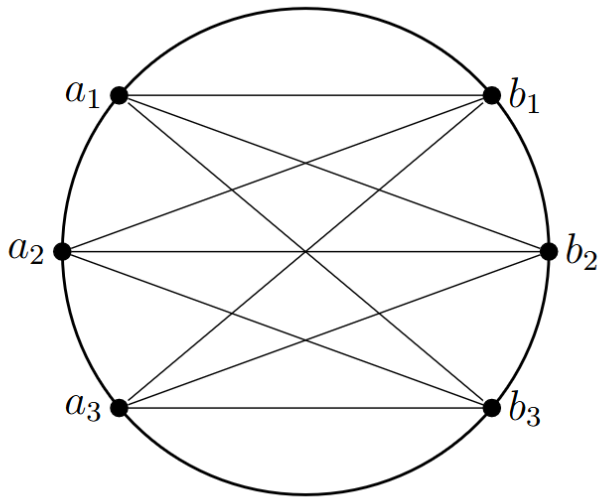


Circle graphs may have high rank-width

Proposition (Hahn, McCarty, Poulsen Nautrup, C., 2026)

The $m \times m$ comparability grid has rank-width at least $m/4$. (polynomial in the number of qubits)





Figures: Rose McCarty, PhD Thesis

Simulation of circle graphs

Circle graphs may have high rank-width.

Simulation of circle graphs

Circle graphs may have high rank-width.

However:

Theorem (Harrison, Iyer, Parekh, Thompson, Zhao, 2025)

Measurement-based quantum computing on **circle graph states** *is efficiently simulable on a classical computer.*

What is the common denominator?

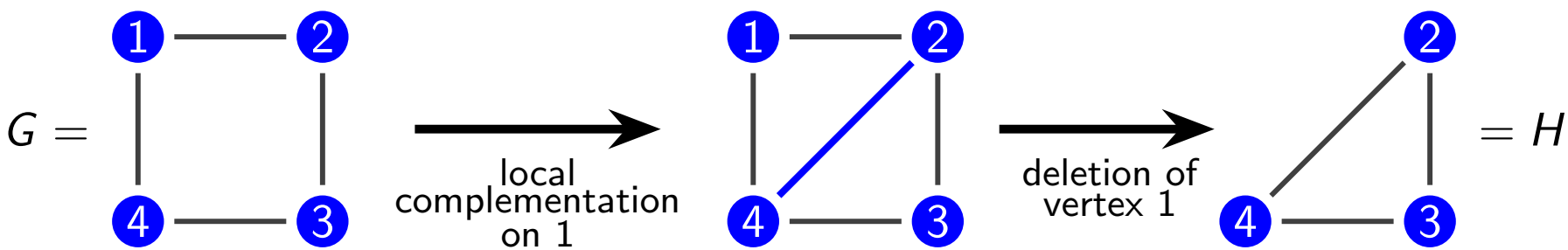
The McCarty-Geelen simulation conjecture

Conjecture (The McCarty-Geelen simulation conjecture)

vertex-minor-closed family *of graphs* $\Longrightarrow$ *efficient classical simulation.*

Vertex-minors

Vertex-minor = local complementation + vertex deletion



Vertex-minor closure

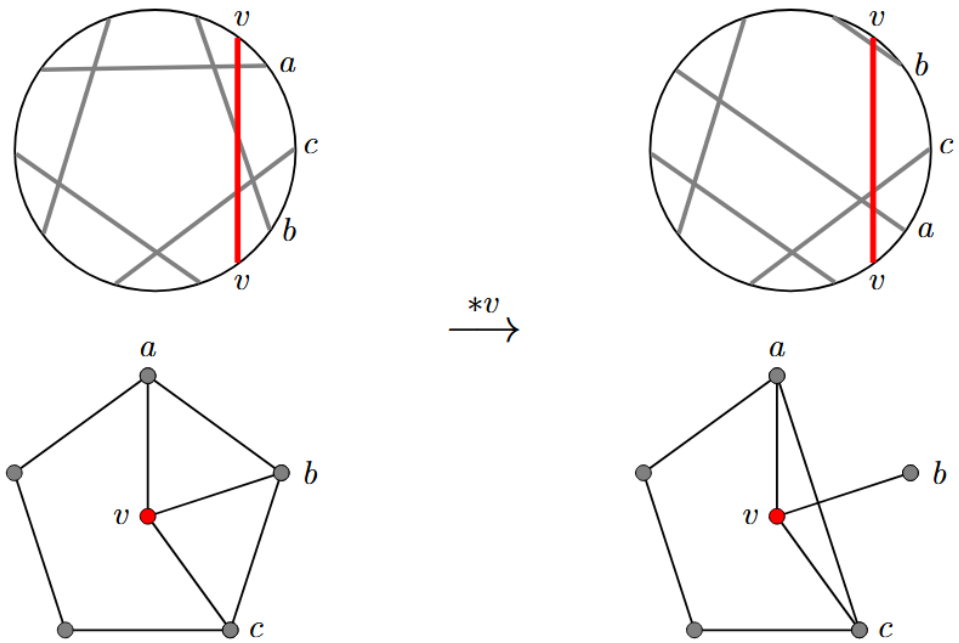
Vertex-minor-closed family of graphs

$=$

any vertex-minor of a graph in the family is also in the family

Circle graphs

Vertex deletion: removing a chord from the chord diagram.
Local complementation: "flipping" some chords.



Proposition

Circle graphs are a proper vertex-minor closed class of graphs.

Figures: Rose McCarty, PhD Thesis

Bounded rank-width

Rank-width is non-increasing under the vertex-minor relation. $\Longrightarrow$

Proposition

For a fixed integer k, graphs of rank-width at most k form a vertex-minor closed class of graphs.

Conjecture (The McCarty-Geelen simulation conjecture)

vertex-minor-closed family *of graphs* $\Longrightarrow$ *efficient classical simulation.*

Classifying graph states according to their entanglement

When do two graph states have the same entanglement ?

"Having the same entanglement" $=$?

When do two graph states have the same entanglement ?

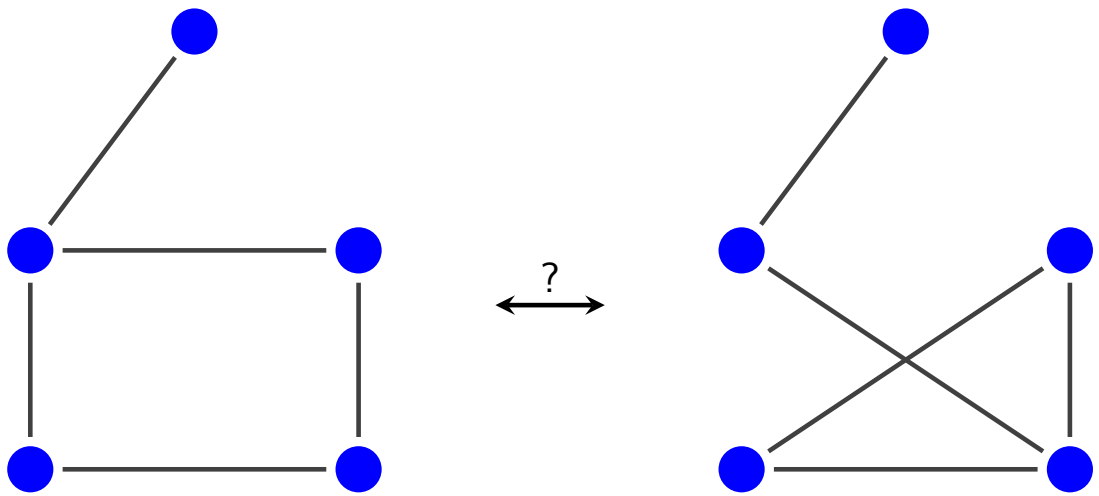
"Having the same entanglement" $=$?

For graph states : LU-equivalence (**L**ocal **U**nitary)

When do two graph states have the same entanglement ?

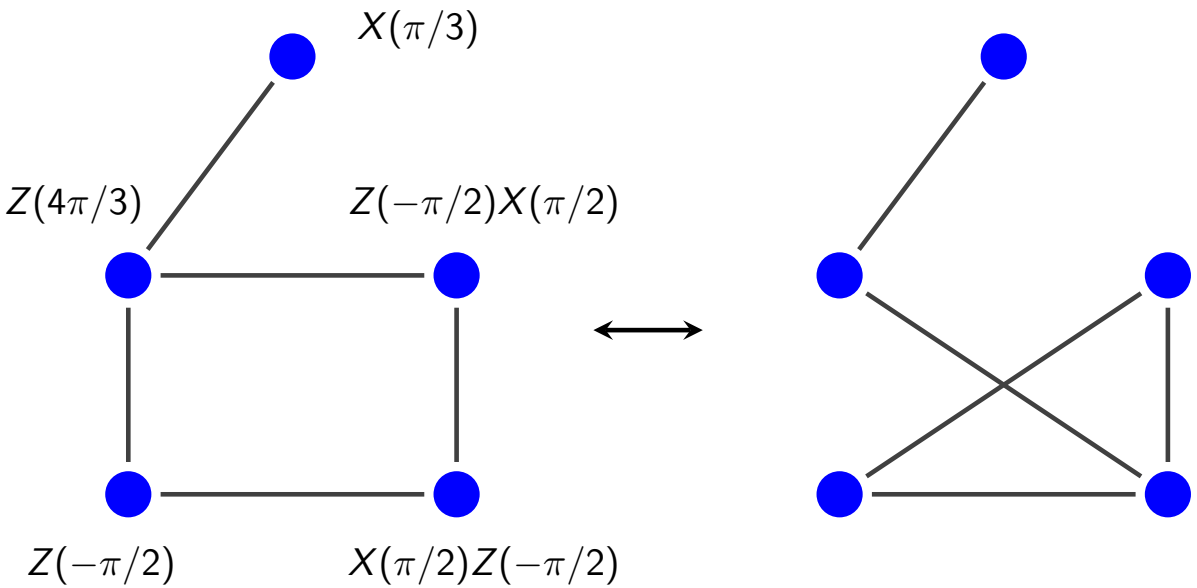
"Having the same entanglement" $=$?

For graph states : LU-equivalence (**L**ocal **U**nitary)



When do two graph states have the same entanglement ?

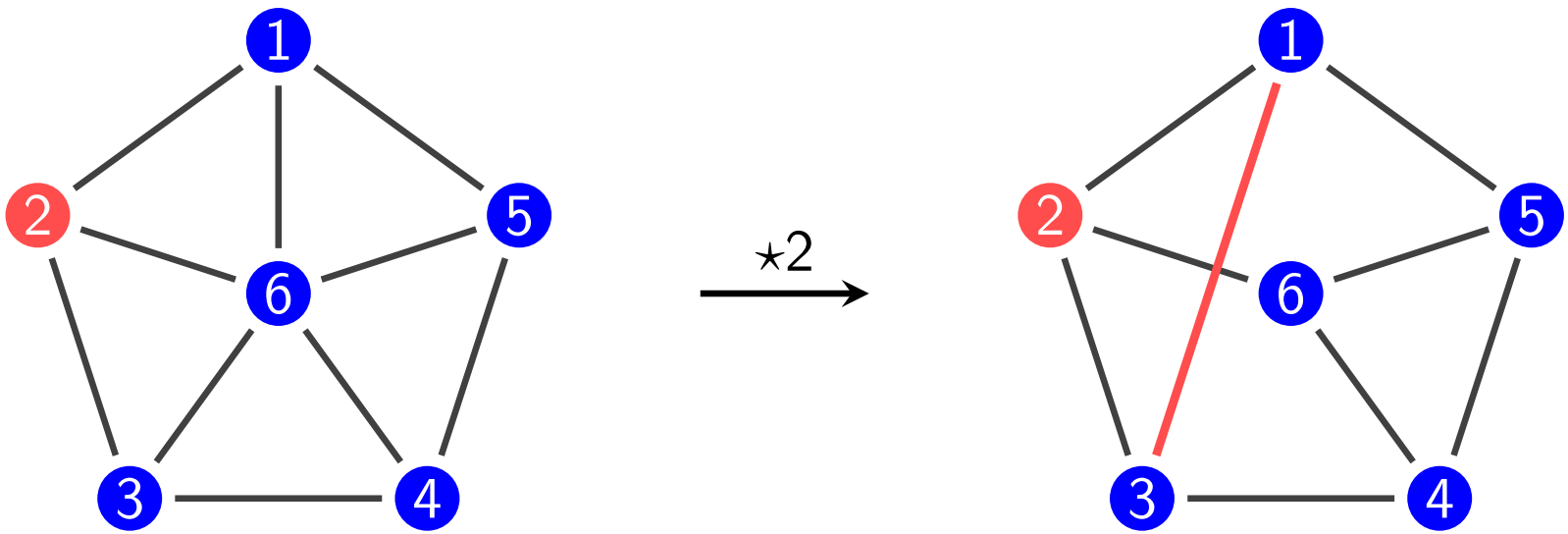
"Having the same entanglement" $=$?
For graph states : LU-equivalence (**L**ocal **U**nitary)



Local complementation and local unitary equivalence

Proposition

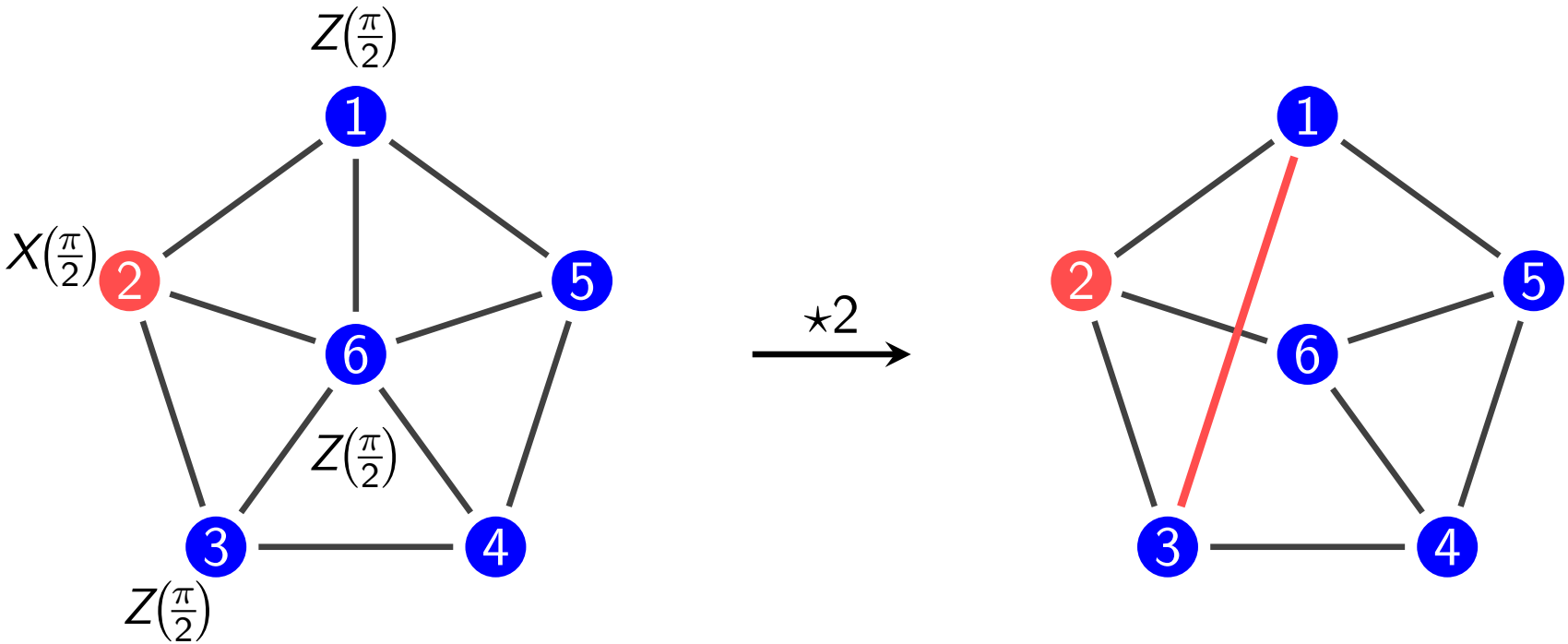
Related by local complementations $\Longrightarrow$ *local unitary equivalent.*



Local complementation and local unitary equivalence

Proposition

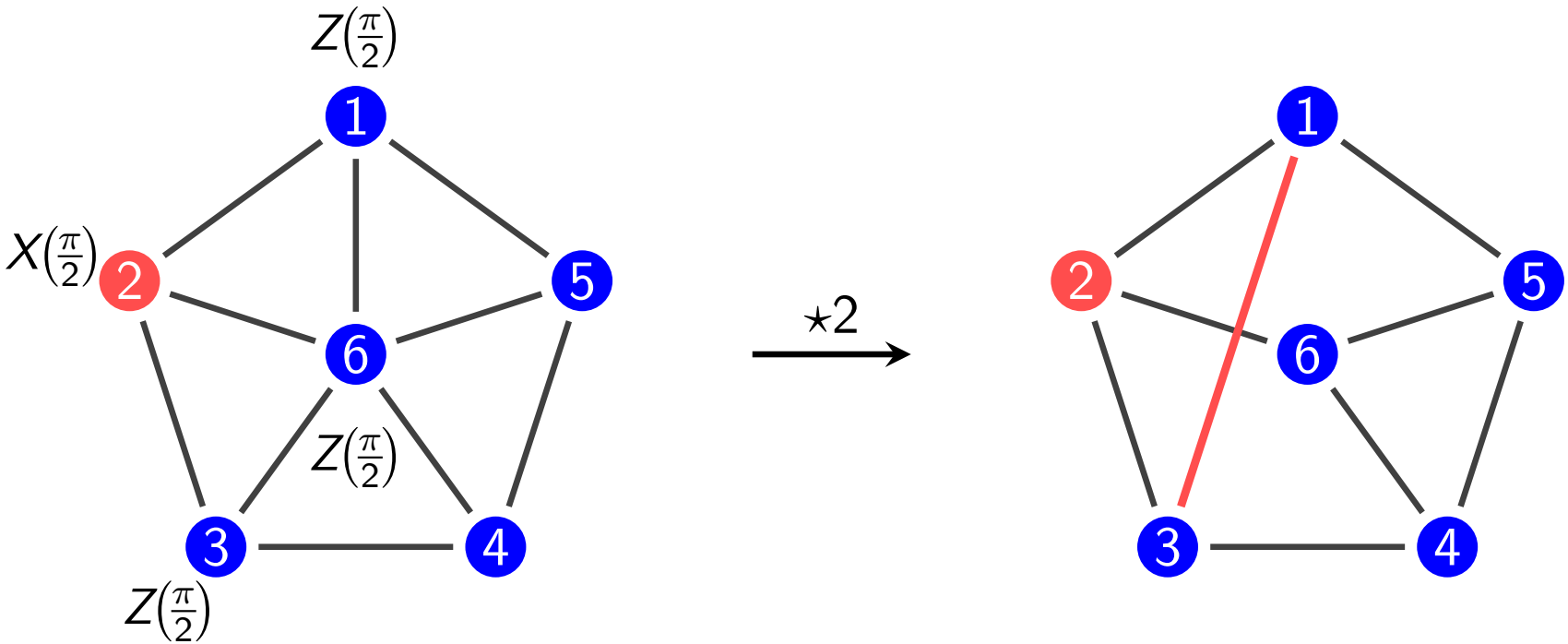
Related by local complementations $\Longrightarrow$ *local unitary equivalent.*



Local complementation and local unitary equivalence

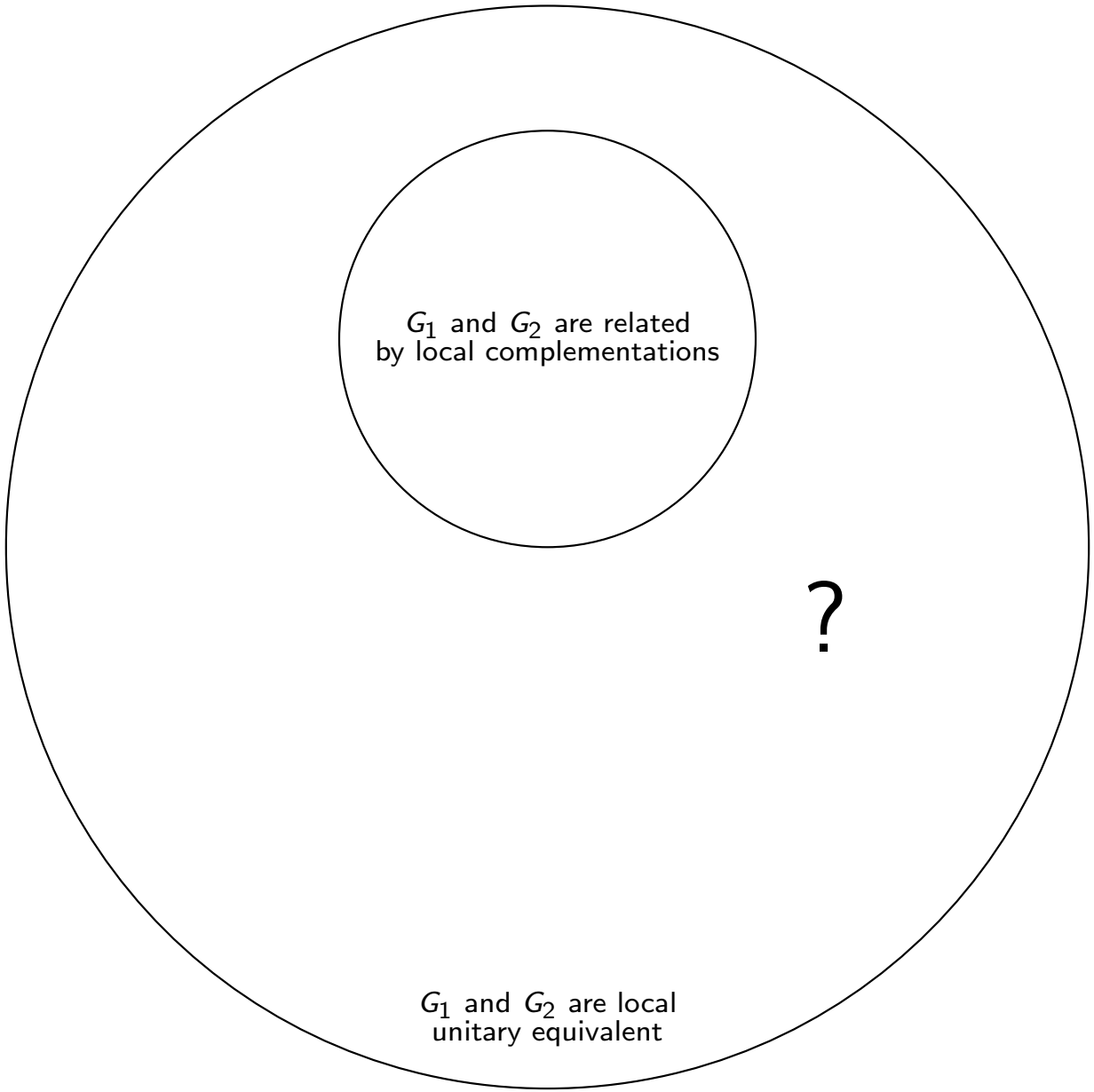
Proposition

Related by local complementations $\Longrightarrow$ *local unitary equivalent.*



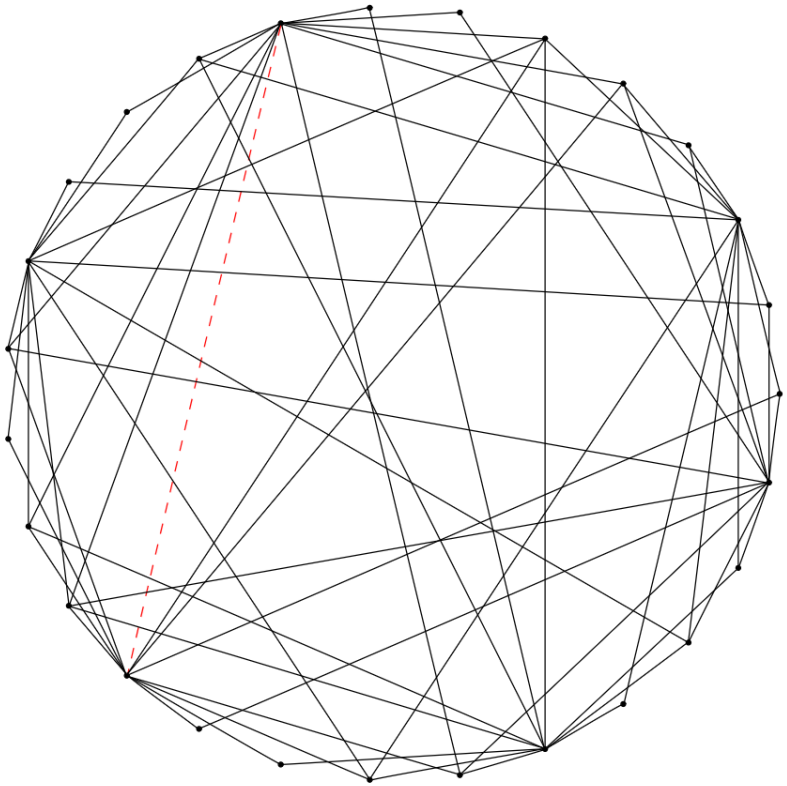
Is the converse true?

Hierarchy


G_1 and G_2 are related
by local complementations
?
G_1 and G_2 are local
unitary equivalent

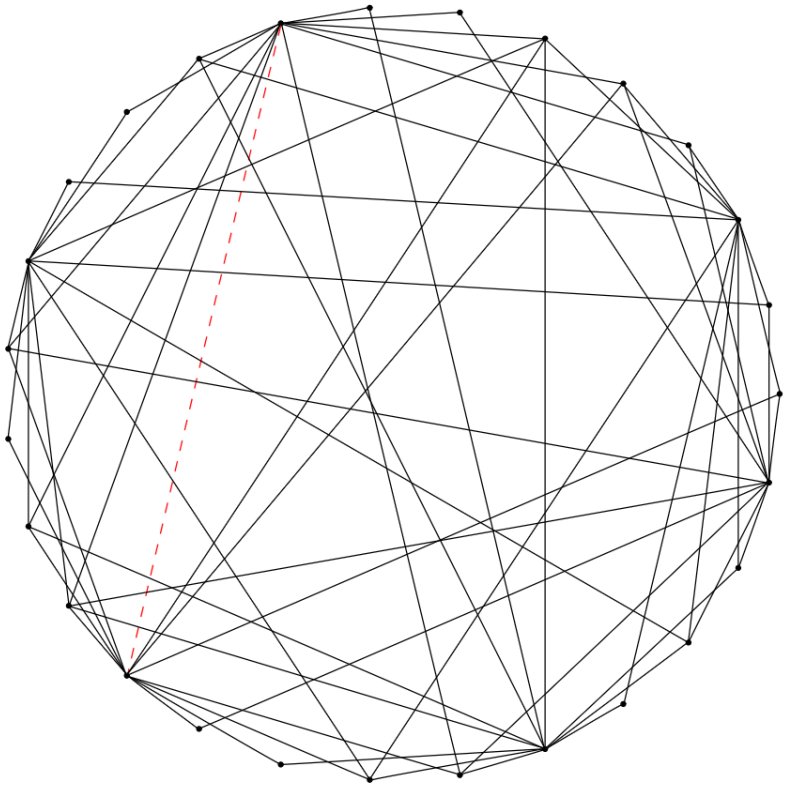
A counter-example

A pair of 27-qubit local unitary equivalent graph states, not related by local complementations (Ji et al. 2007):



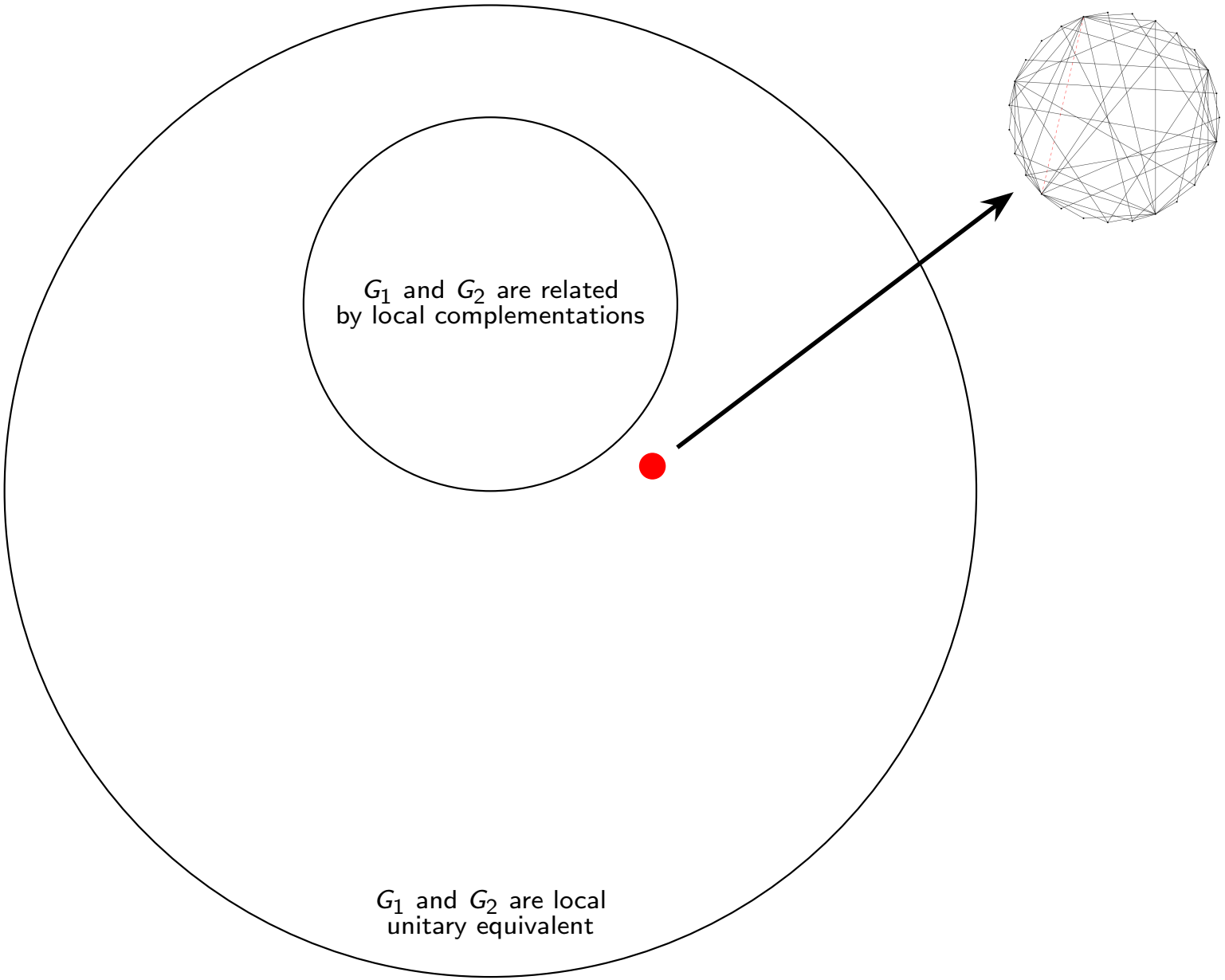
A counter-example

A pair of 27-qubit local unitary equivalent graph states, not related by local complementations (Ji et al. 2007):



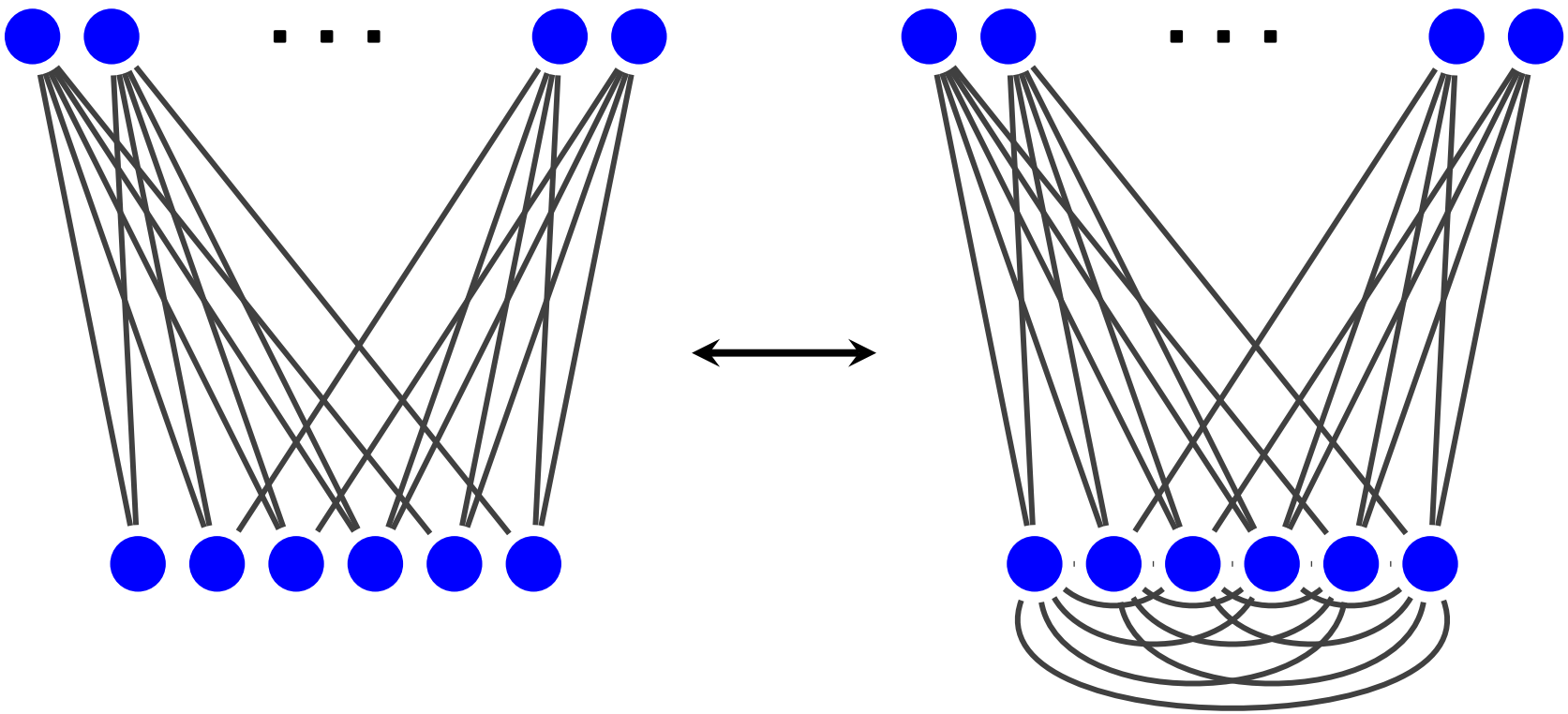
This counter-example is minimal (C. 2026).

Hierarchy


G_1 and G_2 are related
by local complementations
G_1 and G_2 are local
unitary equivalent

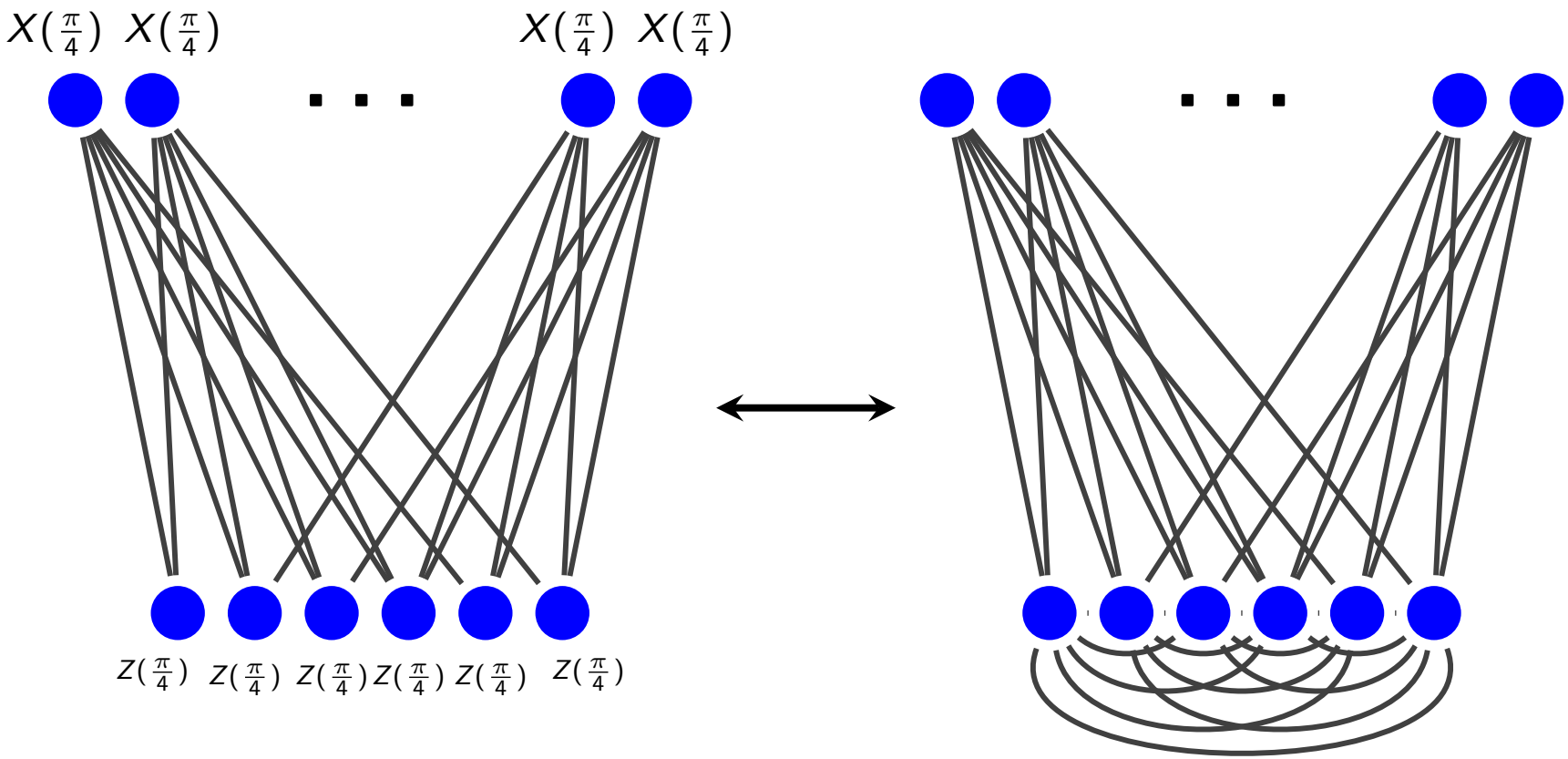
Another look at the 27-qubit counterexample

Equivalent counterexample (Tsimakuridze, Gühne, 2017).



Another look at the 27-qubit counterexample

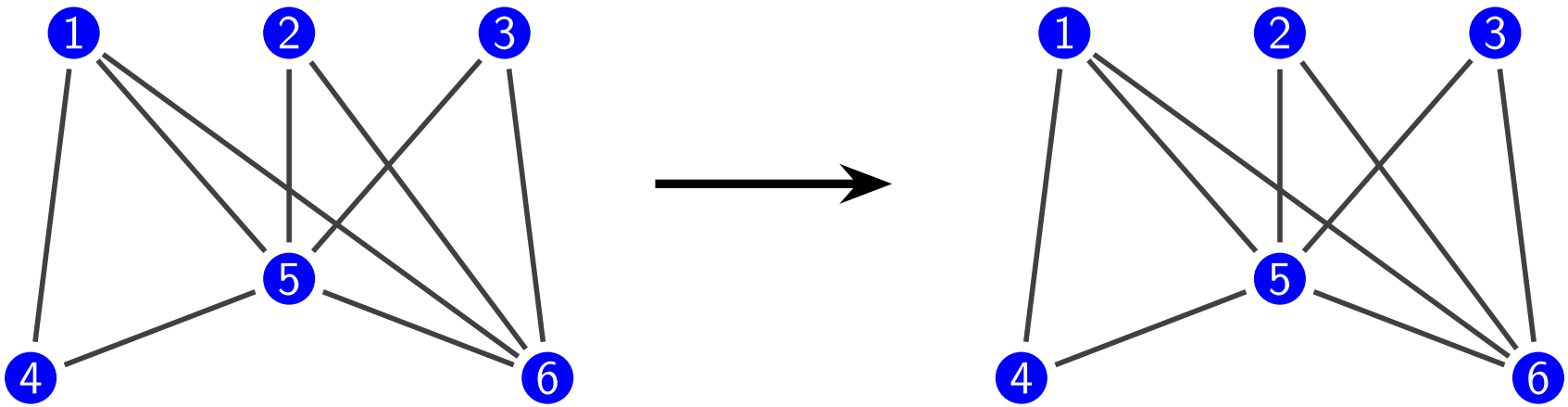
Equivalent counterexample (Tsimakuridze, Gühne, 2017).



Generalizing local complementation to capture local unitary equivalence

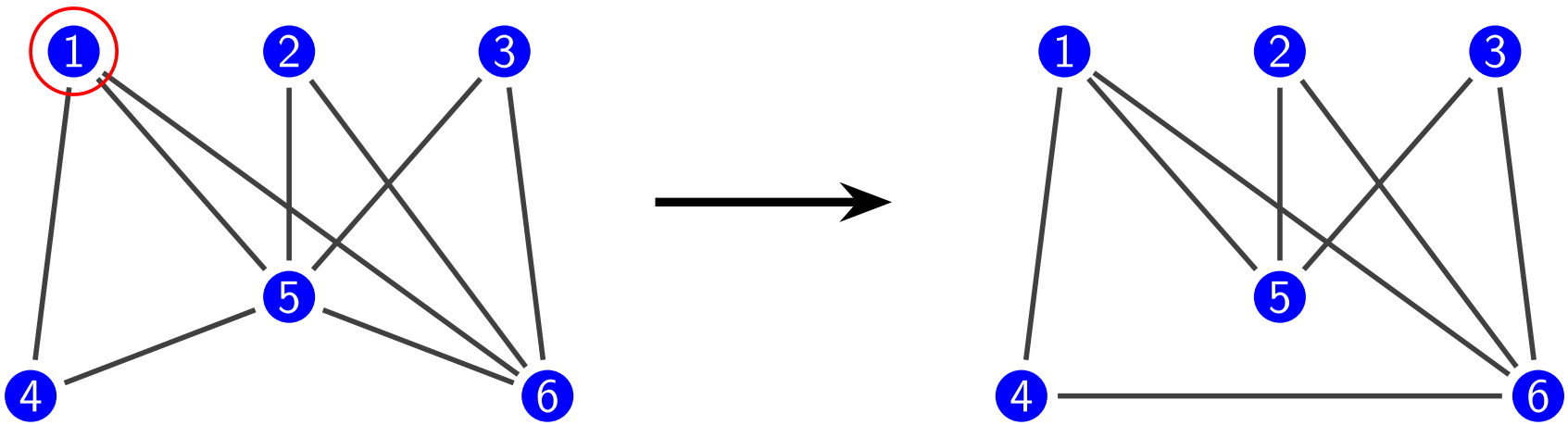
A refinement of idempotent local complementations

A sequence of local complementations may leave the graph invariant.



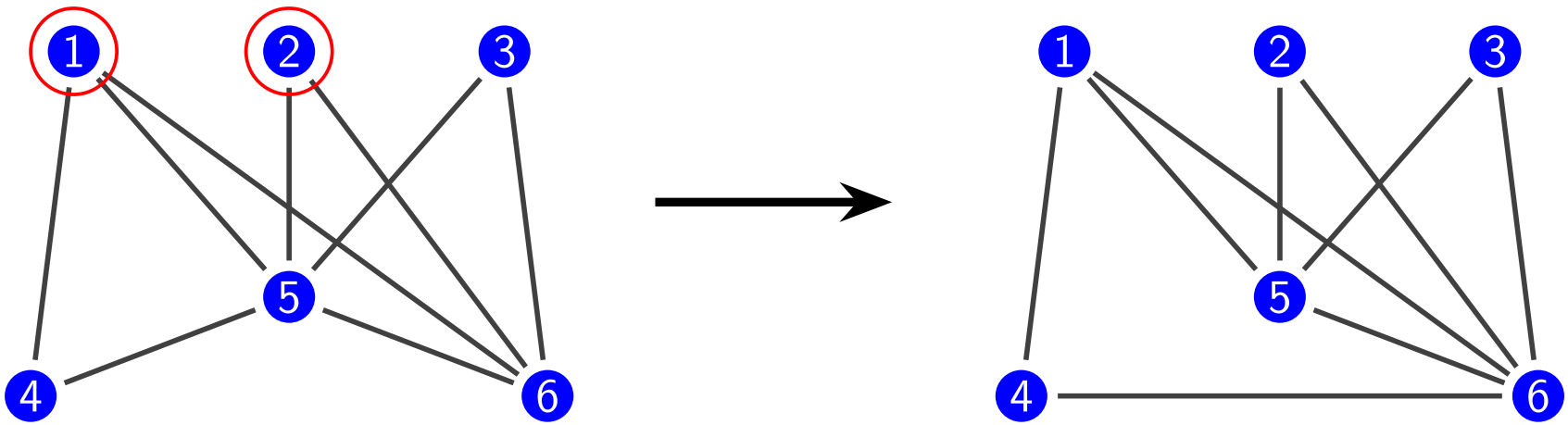
A refinement of idempotent local complementations

A sequence of local complementations may leave the graph invariant.



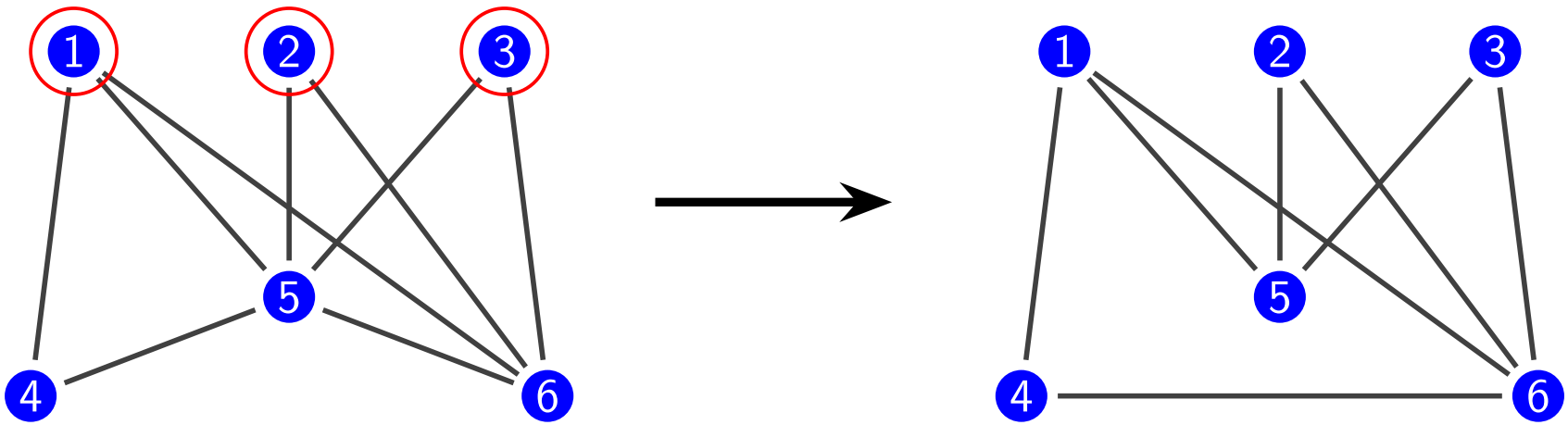
A refinement of idempotent local complementations

A sequence of local complementations may leave the graph invariant.



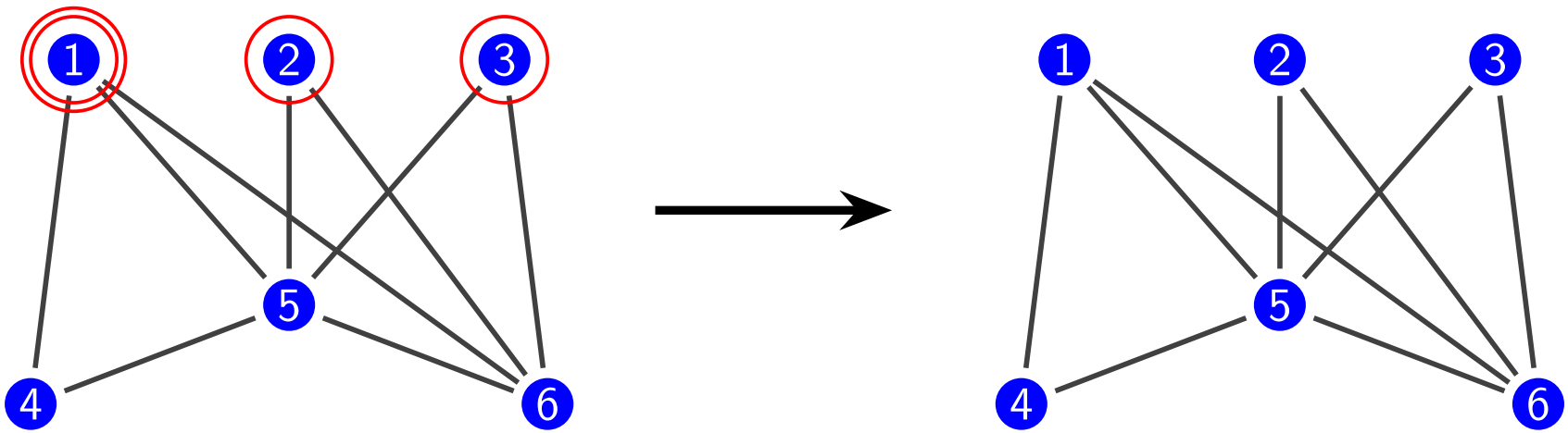
A refinement of idempotent local complementations

A sequence of local complementations may leave the graph invariant.



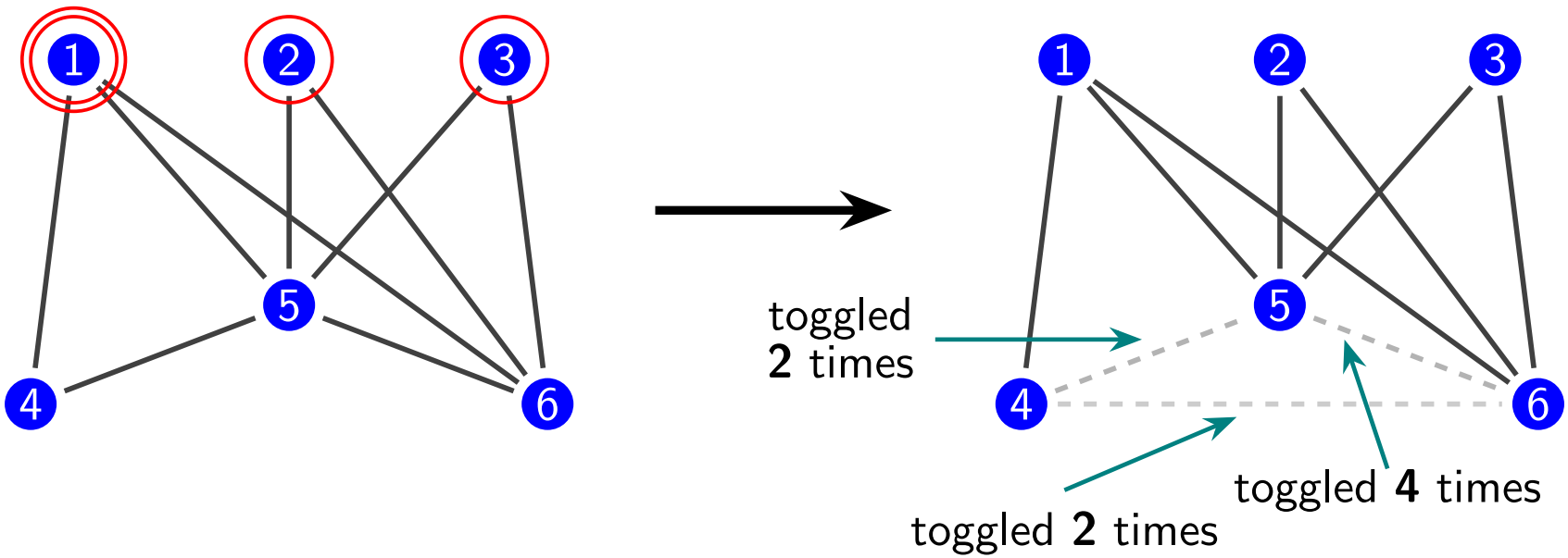
A refinement of idempotent local complementations

A sequence of local complementations may leave the graph invariant.



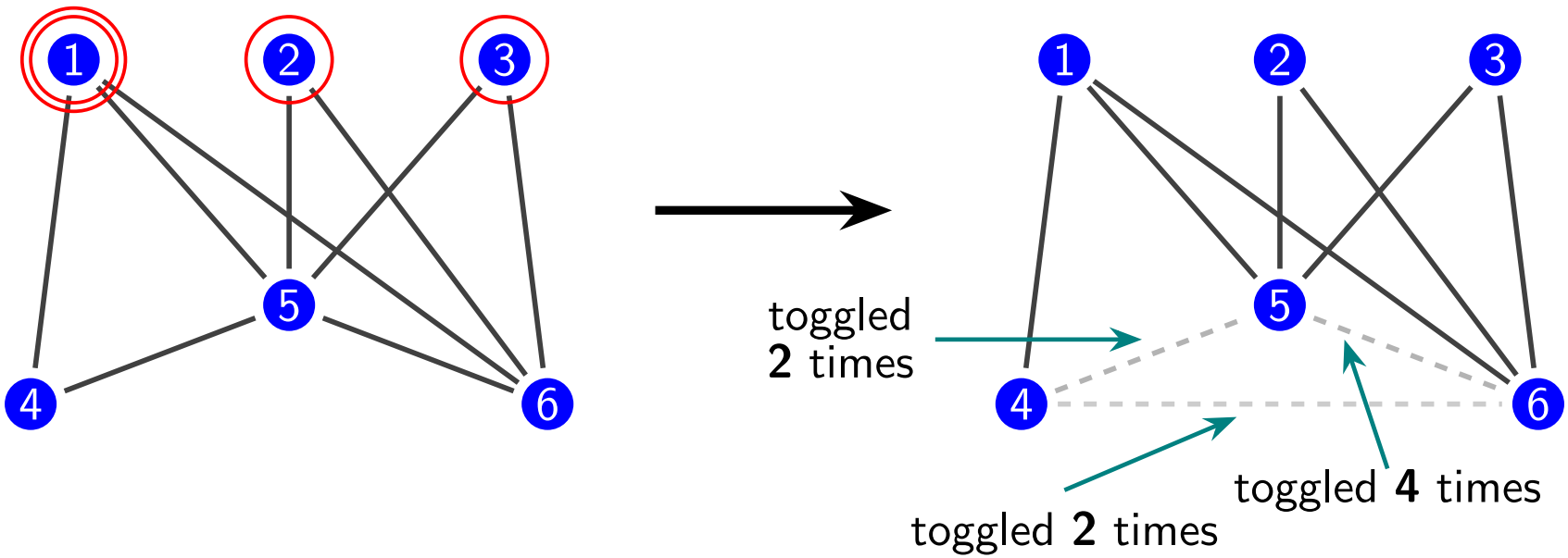
A refinement of idempotent local complementations

A sequence of local complementations may leave the graph invariant.



A refinement of idempotent local complementations

A sequence of local complementations may leave the graph invariant.

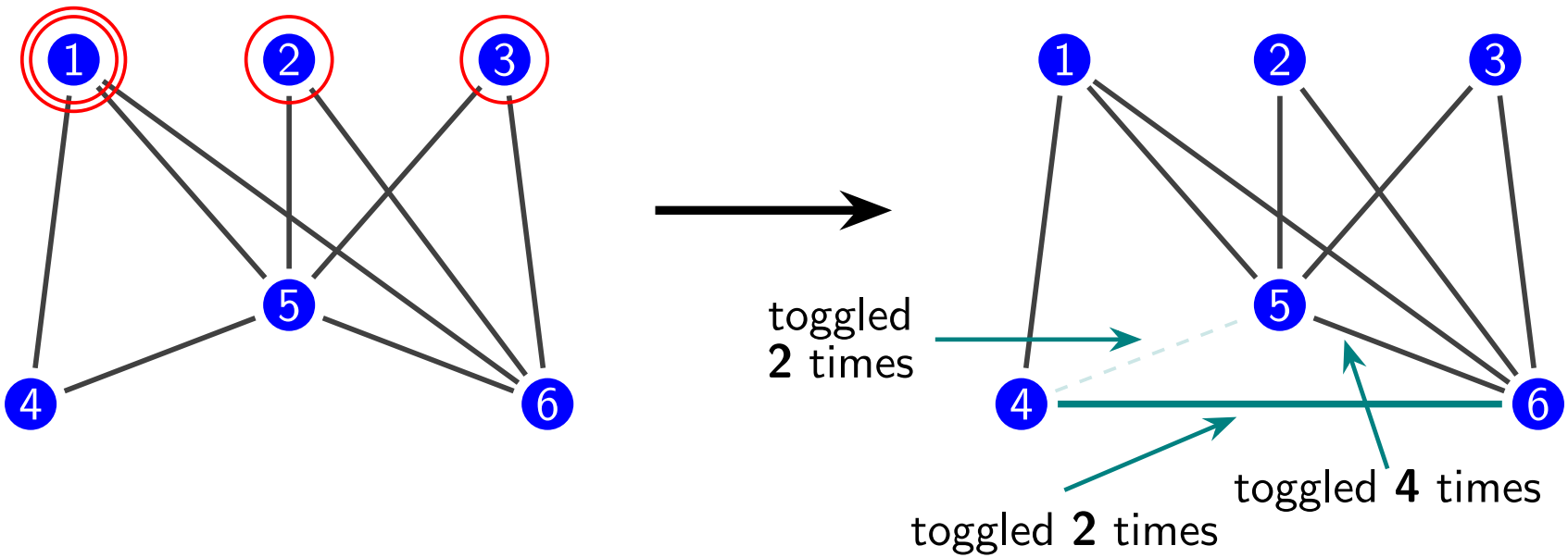


A **2-local complementation** consists in toggling every edge that was toggled 2 mod 4 times by the idempotent local complementations.

+ additional conditions on the edges

A refinement of idempotent local complementations

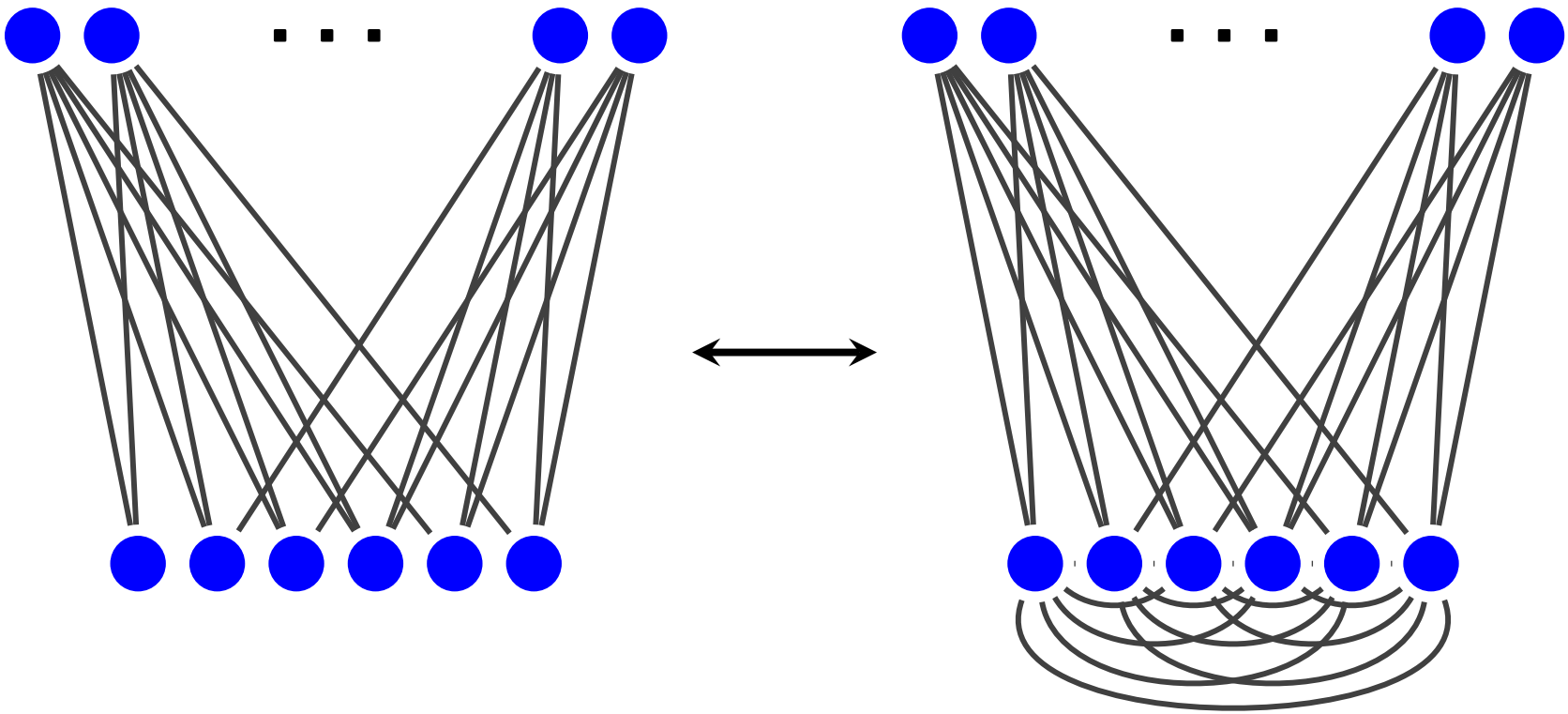
A sequence of local complementations may leave the graph invariant.



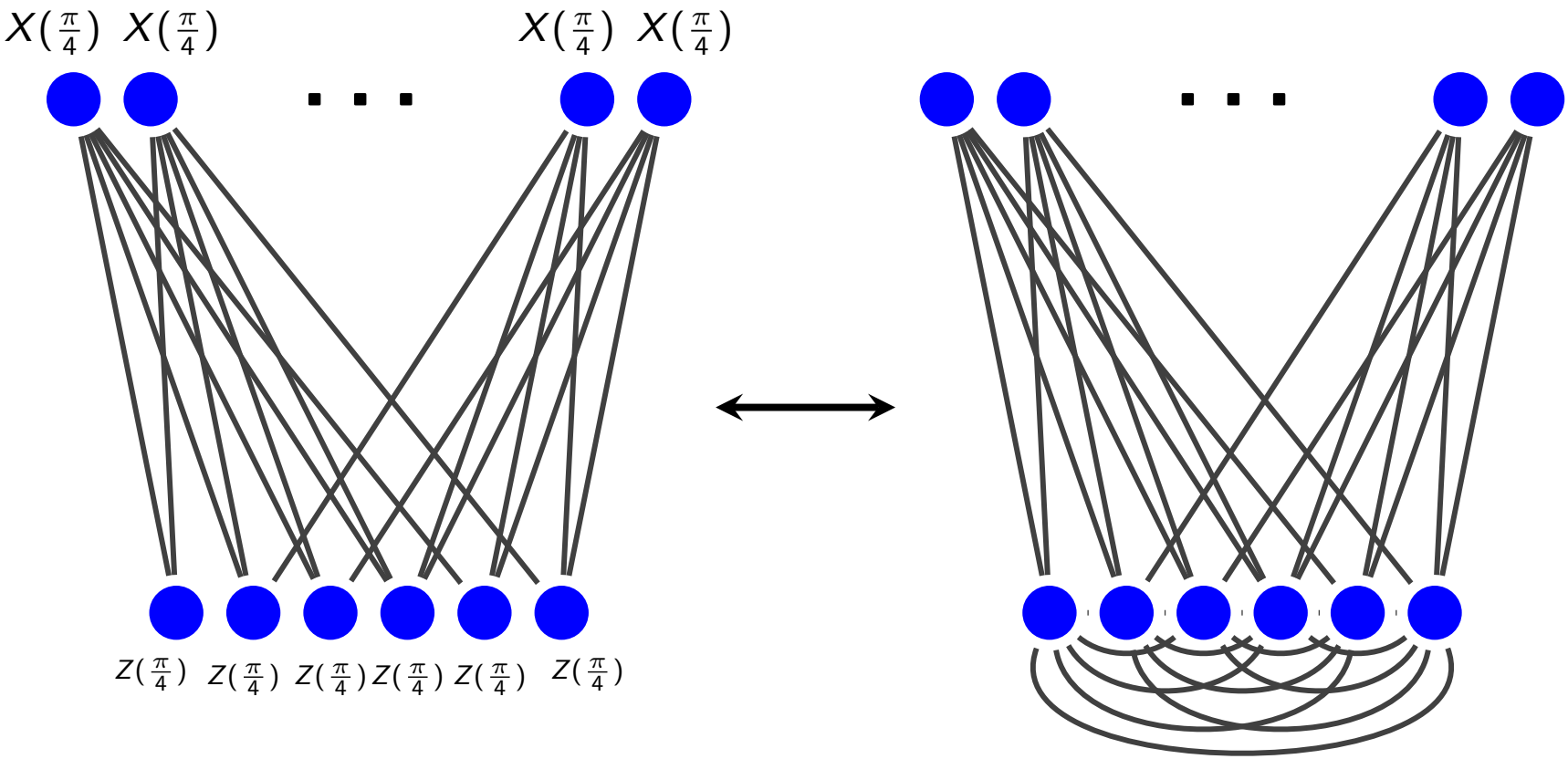
A **2-local complementation** consists in toggling every edge that was toggled 2 mod 4 times by the idempotent local complementations.

+ additional conditions on the edges

Example of a 2-local complementation



Example of a 2-local complementation



r-local complementation

3-local complementation is a refinement of idempotent 2-local complementation, and so on...

$\rightarrow$ Infinite family of graphical operations parametrised by an integer r:

r-local complementations

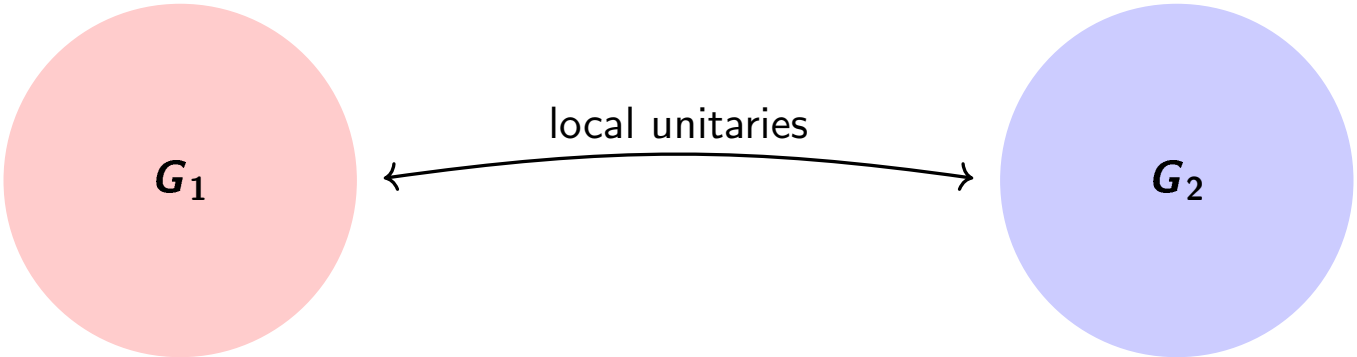
1-local complementation = local complementation.

r-local complementation captures local unitary equivalence

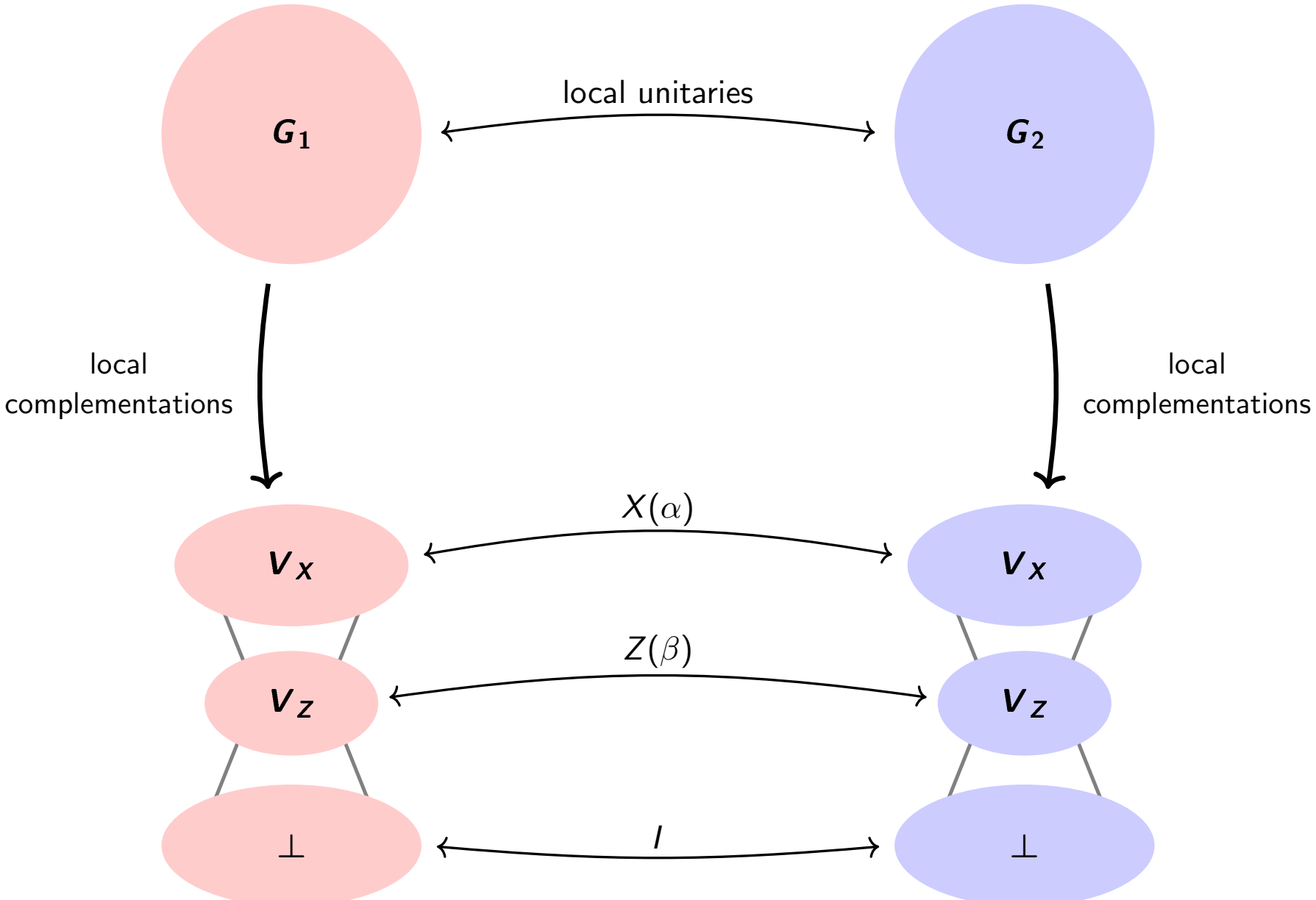
Theorem (C, Perdrix, 2025)

Two graph states are local unitary equivalent if and only if the two underlying graphs are related by a sequence of r-local complementations.

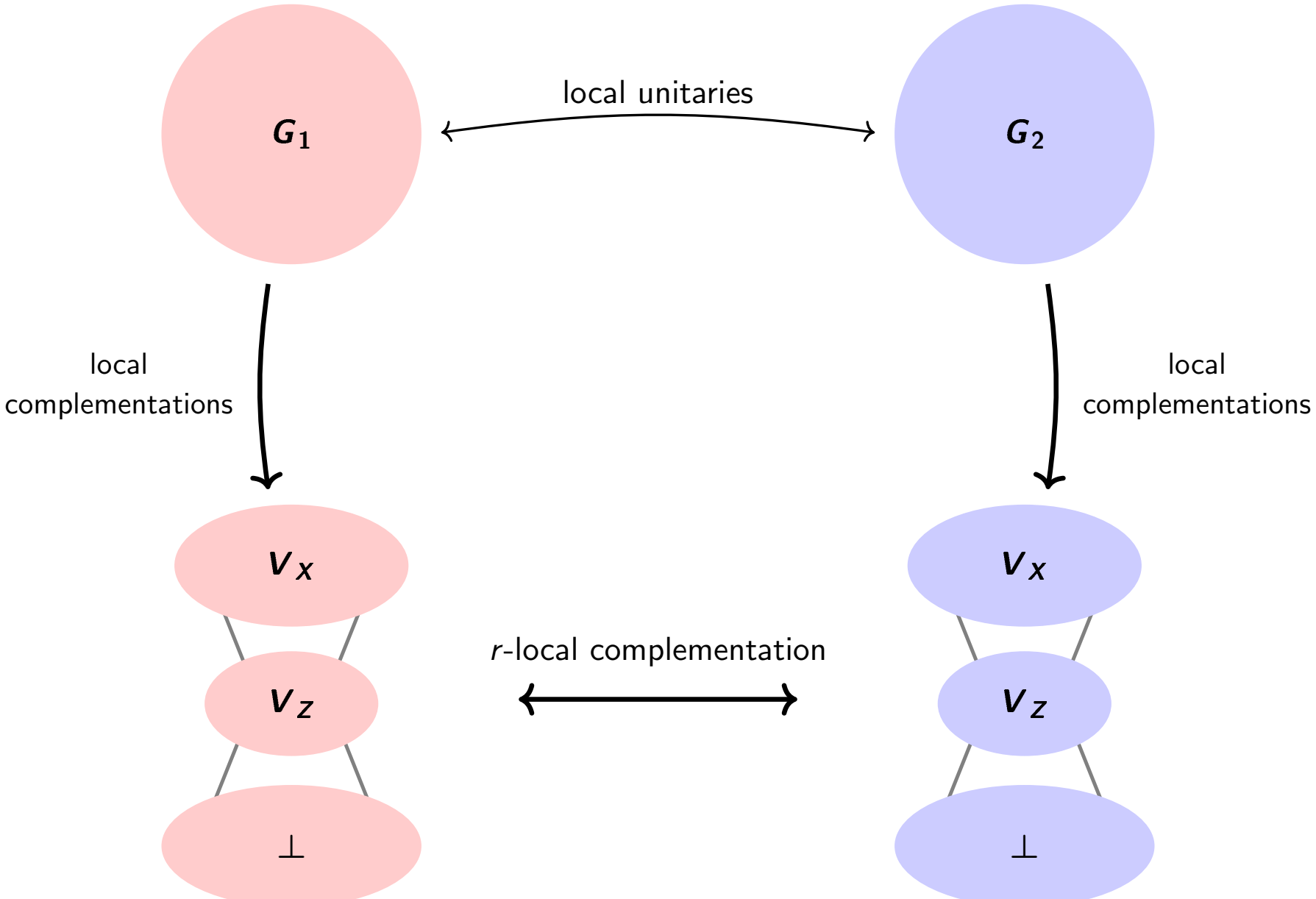
Proof sketch: Standard form for graph states



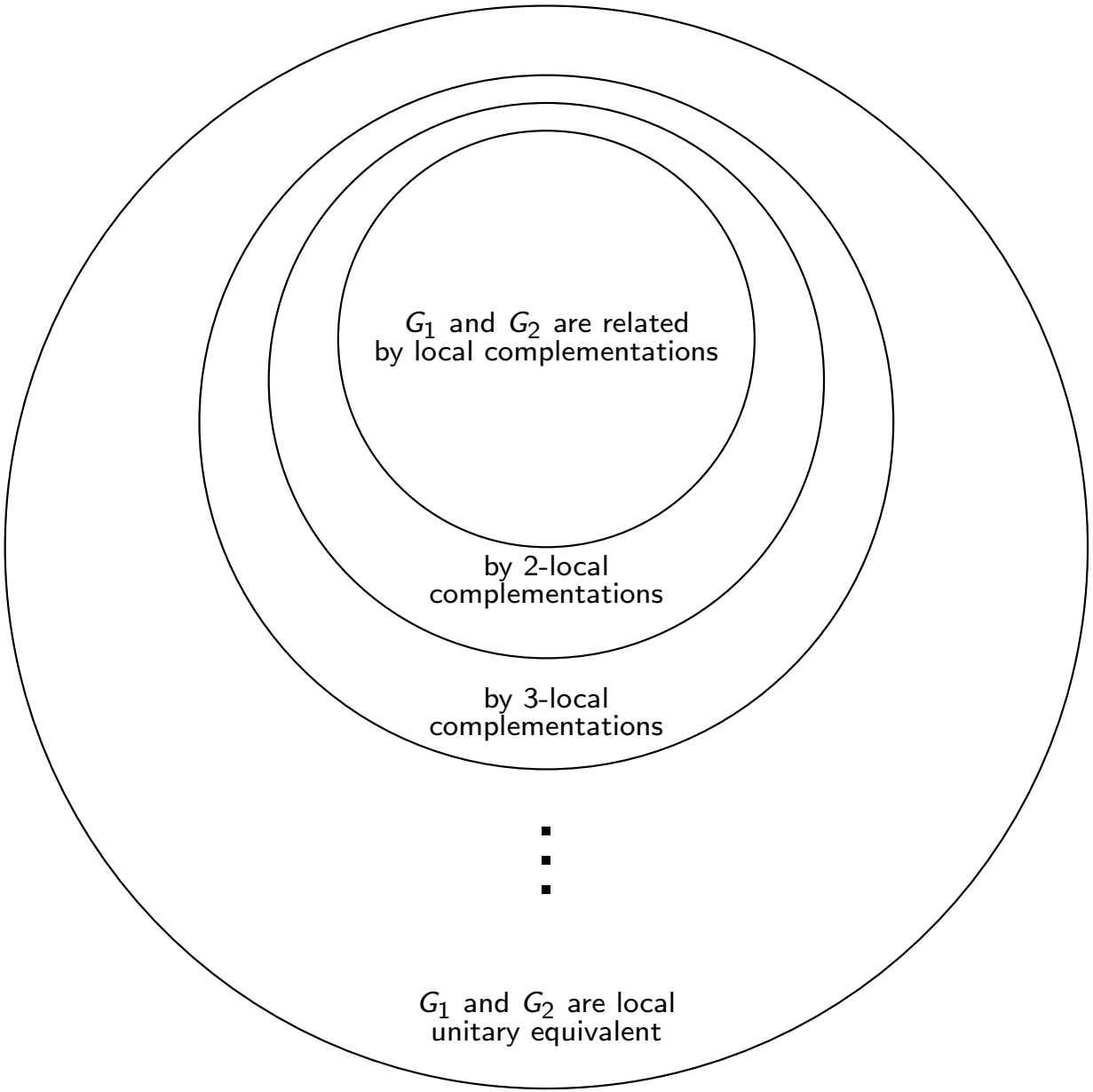
Proof sketch: Standard form for graph states



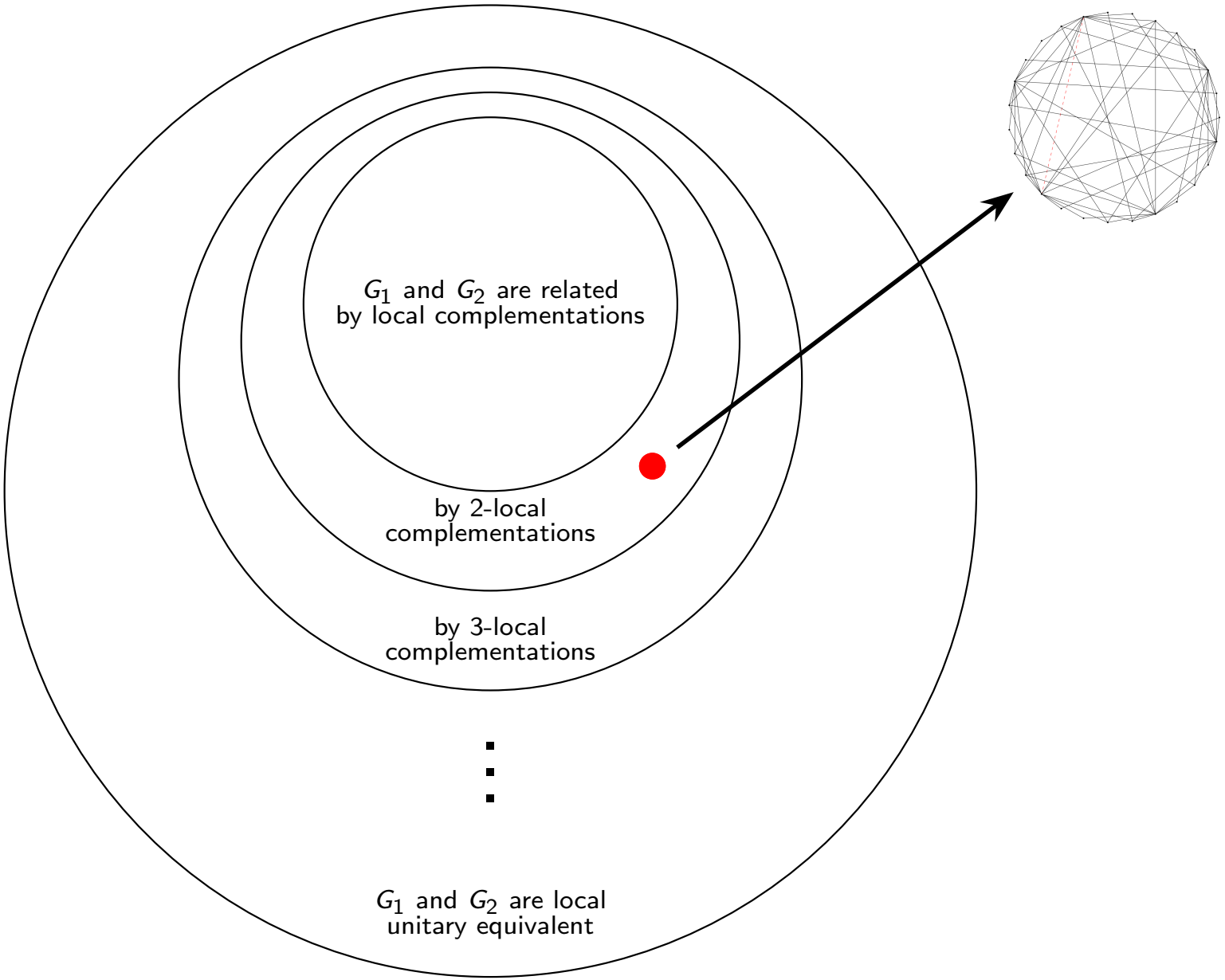
Proof sketch: Standard form for graph states



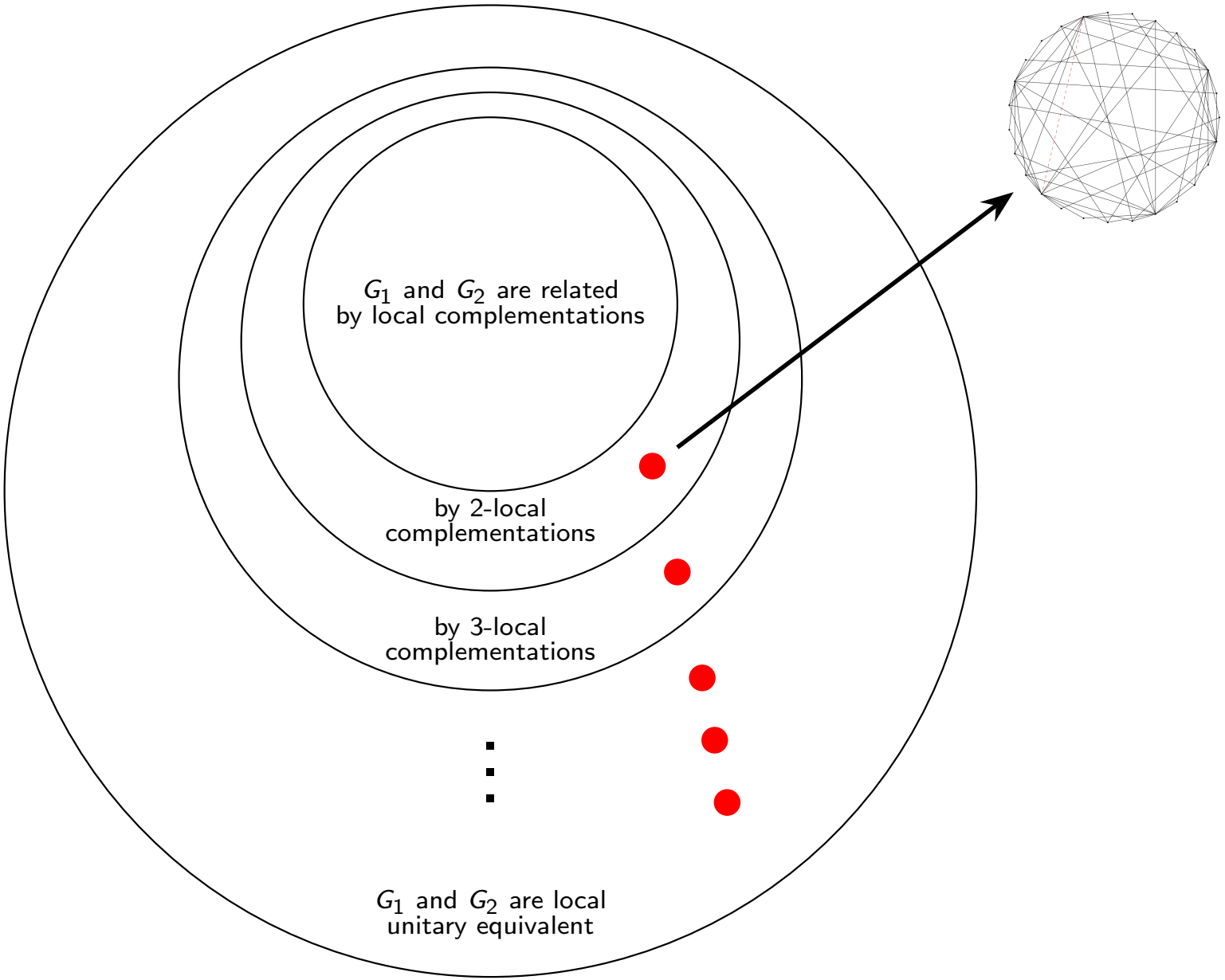
An infinite hierarchy of local equivalences



An infinite hierarchy of local equivalences



An infinite hierarchy of local equivalences



Summary

Summary

1 - Classifying graph states according to computational power as a quantum resource

Summary

1 - Classifying graph states according to computational power as a quantum resource

- Good resources (e.g. grid graph states)

Summary

1 - Classifying graph states according to computational power as a quantum resource

- Good resources (e.g. grid graph states)
- Bad resources (e.g. low rank-width graph states, circle graph states)

Summary

1 - Classifying graph states according to computational power as a quantum resource

- Good resources (e.g. grid graph states)
- Bad resources (e.g. low rank-width graph states, circle graph states)

Future work: McCarty-Geelen simulation conjecture.

Summary

1 - Classifying graph states according to computational power as a quantum resource

- Good resources (e.g. grid graph states)
- Bad resources (e.g. low rank-width graph states, circle graph states)

Future work: McCarty-Geelen simulation conjecture.

2 - Classifying graph states according to their entanglement

Summary

1 - Classifying graph states according to computational power as a quantum resource

- Good resources (e.g. grid graph states)
- Bad resources (e.g. low rank-width graph states, circle graph states)

Future work: McCarty-Geelen simulation conjecture.

2 - Classifying graph states according to their entanglement

- r-local complementation captures LU-equivalence of graph states

Summary

1 - Classifying graph states according to computational power as a quantum resource

- Good resources (e.g. grid graph states)
- Bad resources (e.g. low rank-width graph states, circle graph states)

Future work: McCarty-Geelen simulation conjecture.

2 - Classifying graph states according to their entanglement

- r-local complementation captures LU-equivalence of graph states
- infinite strict hierarchy of local equivalences between graph states

Summary

1 - Classifying graph states according to computational power as a quantum resource

- Good resources (e.g. grid graph states)
- Bad resources (e.g. low rank-width graph states, circle graph states)

Future work: McCarty-Geelen simulation conjecture.

2 - Classifying graph states according to their entanglement

- r-local complementation captures LU-equivalence of graph states
- infinite strict hierarchy of local equivalences between graph states

Future work: A quantum generalization of the notion of vertex-minor.

Thanks

