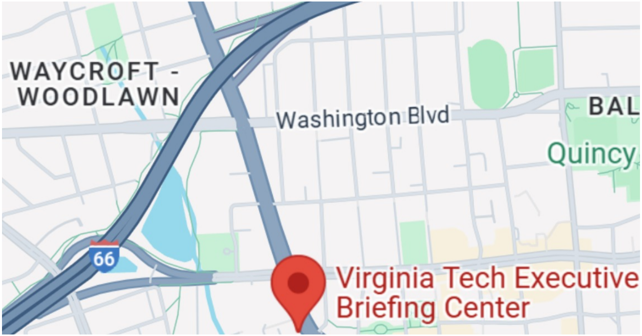
Graph theory tutorial

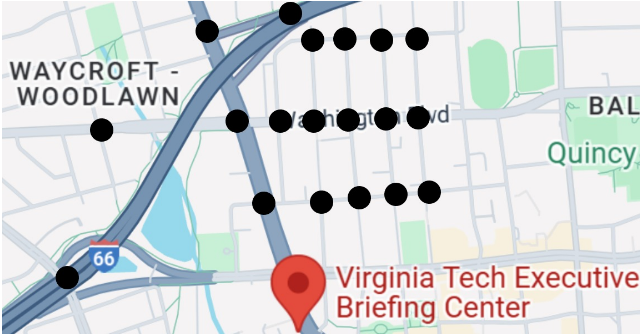
Rose McCarty
School of Math and School of Computer Science
Georgia Tech

May 28, 2025

Classical graph theory

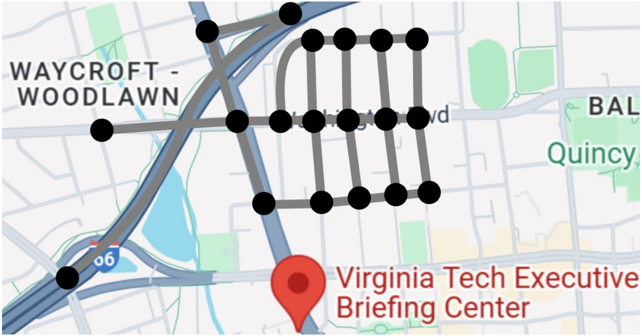


Classical graph theory



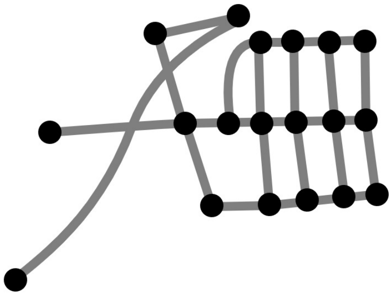
Vertices **V**

Classical graph theory



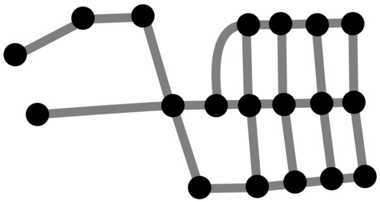
Vertices **V**
Edges **E**

Classical graph theory



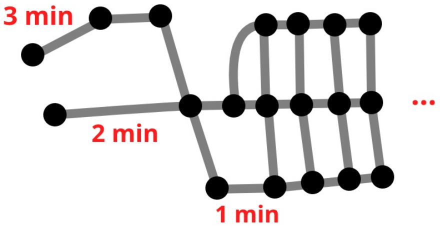
Vertices **V**
Edges **E**
Graph $G = (\mathbf{V}, \mathbf{E})$

Classical graph theory



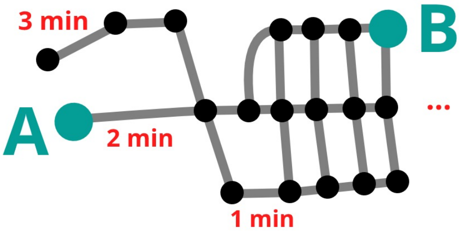
Vertices **V**
Edges **E**
Graph $G = (\mathbf{V}, \mathbf{E})$

Classical graph theory



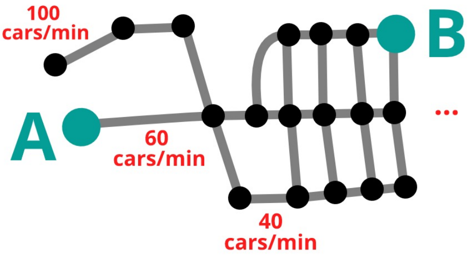
Vertices **V**
Edges **E**
Graph $G = (\mathbf{V}, \mathbf{E})$
Edge-weights $\mathbf{w} : \mathbf{E} \to \mathbb{R}_{\geq 0}$

Classical graph theory



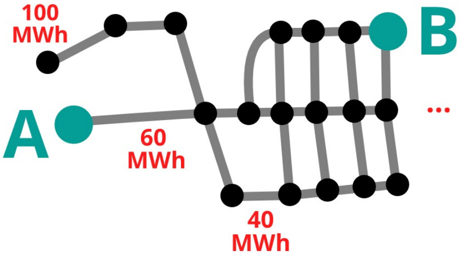
What is the fastest route from **A** to **B**?

Classical graph theory



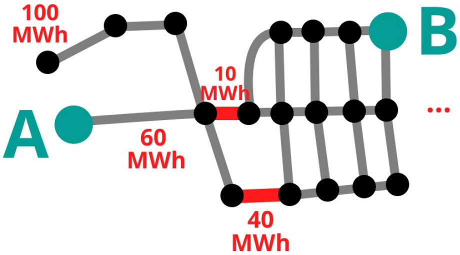
What is the fastest route from **A** to **B**?
How many cars can go from **A** to **B** without traffic?

Classical graph theory



What is the fastest route from **A** to **B**?
How much electricity can **A** send to **B**?

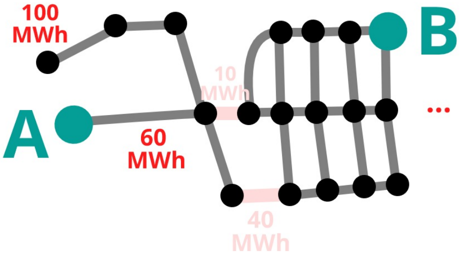
Classical graph theory



Theorem:
max-flow
=
min-cut

What is the fastest route from **A** to **B**?
How much electricity can **A** send to **B**?

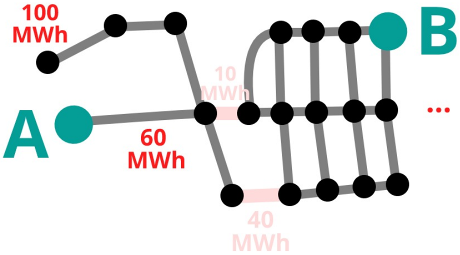
Classical graph theory



Theorem:
max-flow
=
min-cut

What is the fastest route from **A** to **B**?
How much electricity can **A** send to **B**?

Classical graph theory

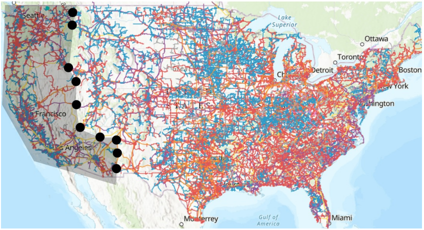


Theorem:
max-flow
=
min-cut

What is the fastest route from **A** to **B**?
How much electricity can **A** send to **B**?
Where is the power grid the most vulnerable?

Classical graph theory

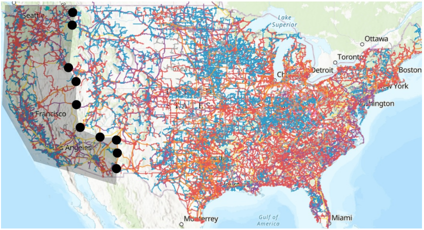
A graph is a **bad expander** if deleting a few vertices/edges can disconnect a huge part.



What is the fastest route from **A** to **B**?
How much electricity can **A** send to **B**?
Where is the power grid the most vulnerable?

Classical graph theory

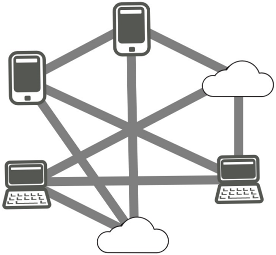
A graph is a **bad expander** if deleting a few vertices/edges can disconnect a huge part.



What is the fastest route from **A** to **B**?
How much electricity can **A** send to **B**?
Where is the power grid the most vulnerable?
How can we design robust networks (ie **good expanders**)?

Classical graph theory

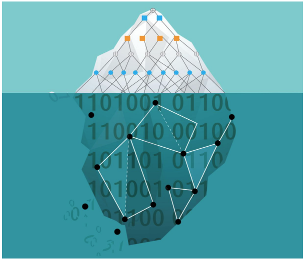
A graph is a **bad expander** if deleting a few vertices/edges can disconnect a huge part.



What is the fastest route from **A** to **B**?
How much electricity can **A** send to **B**?
Where is the power grid the most vulnerable?
How can we design robust networks (ie **good expanders**)?

Classical graph theory

A graph is a **bad expander** if deleting a few vertices/edges can disconnect a huge part.

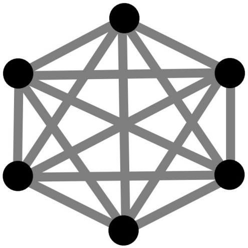


error correction, pseudorandomness, hash functions, Markov chains

What is the fastest route from **A** to **B**?
How much electricity can **A** send to **B**?
Where is the power grid the most vulnerable?
How can we design robust networks (ie **good expanders**)?

Classical graph theory

A graph is a **bad expander** if deleting a few vertices/edges can disconnect a huge part.

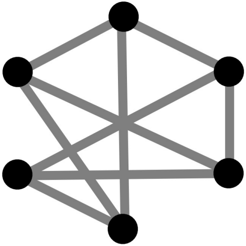


A **clique** is the best expander.

What is the fastest route from **A** to **B**?
How much electricity can **A** send to **B**?
Where is the power grid the most vulnerable?
How can we design robust networks (ie **good expanders**)?

Classical graph theory

A graph is a **bad expander** if deleting a few vertices/edges can disconnect a huge part.



We understand optimal expanders where each vertex is incident to k edges.

What is the fastest route from **A** to **B**?
How much electricity can **A** send to **B**?
Where is the power grid the most vulnerable?
How can we design robust networks (ie **good expanders**)?

Classical graph theory

A graph is a **bad expander** if deleting a few vertices/edges can disconnect a huge part.

Theorem [LT]: Road networks without bridges are bad expanders.

What is the fastest route from **A** to **B**?
How much electricity can **A** send to **B**?
Where is the power grid the most vulnerable?
How can we design robust networks (ie **good expanders**)?

Classical graph theory

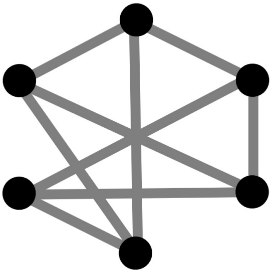
A graph is a **bad expander** if deleting a few vertices/edges can disconnect a huge part.

Theorem [LT]: Road networks without bridges are bad expanders.

What is the fastest route from **A** to **B**?
How much electricity can **A** send to **B**?
Where is the power grid the most vulnerable?
How can we design robust networks (ie **good expanders**)?

Keywords:
connectivity, separators, submodularity, linear programming

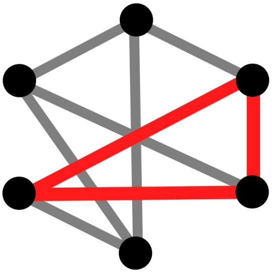
Modern trends in graph theory



1D-expander

How can we design **high-dimensional** expanders?

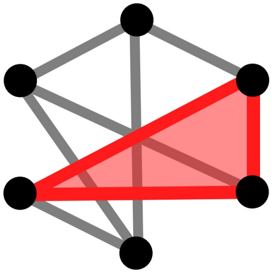
Modern trends in graph theory



2D-expander

How can we design **high-dimensional** expanders?

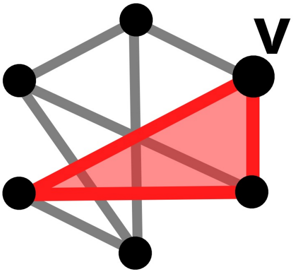
Modern trends in graph theory



2D-expander

How can we design **high-dimensional** expanders?

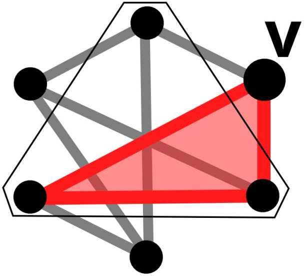
Modern trends in graph theory



2D-expander

How can we design **high-dimensional** expanders?

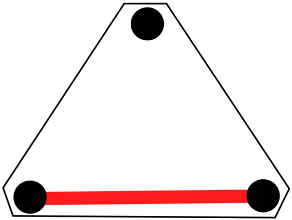
Modern trends in graph theory



2D-expander

How can we design **high-dimensional** expanders?

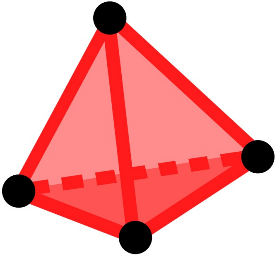
Modern trends in graph theory



2D-expander

How can we design **high-dimensional** expanders?

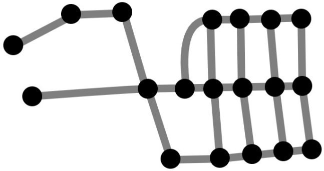
Modern trends in graph theory



3D-expander

How can we design **high-dimensional** expanders?

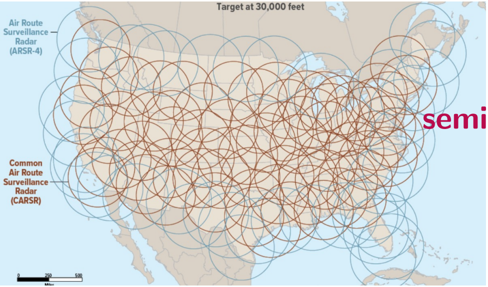
Modern trends in graph theory



planar

How can we design **high-dimensional** expanders?
How can we determine and utilize **structural** properties?

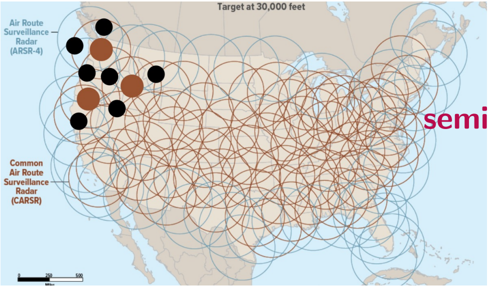
Modern trends in graph theory



semi-algebraic

How can we design **high-dimensional** expanders?
How can we determine and utilize **structural** properties?

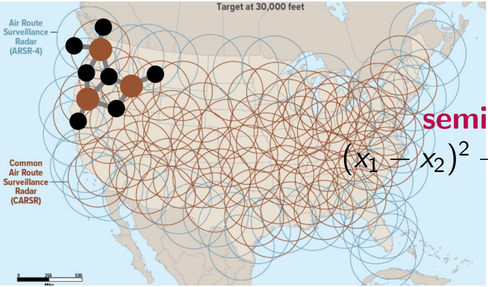
Modern trends in graph theory



semi-algebraic

How can we design **high-dimensional** expanders?
How can we determine and utilize **structural** properties?

Modern trends in graph theory

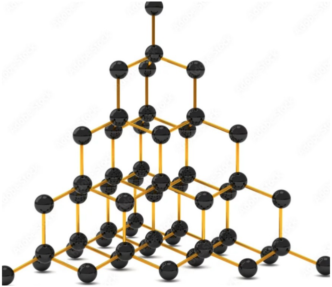


semi-algebraic

$(x_1 - x_2)^2 + (y_1 - y_2)^2 \leq r^2$

How can we design **high-dimensional** expanders?
How can we determine and utilize **structural** properties?

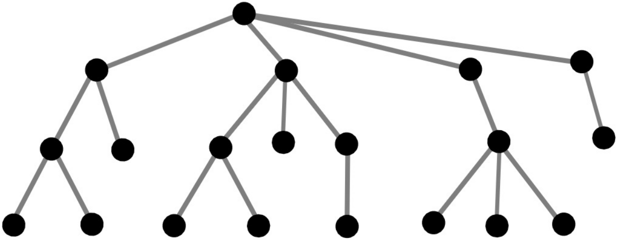
Modern trends in graph theory



lattice
(diamond)

How can we design **high-dimensional** expanders?
How can we determine and utilize **structural** properties?

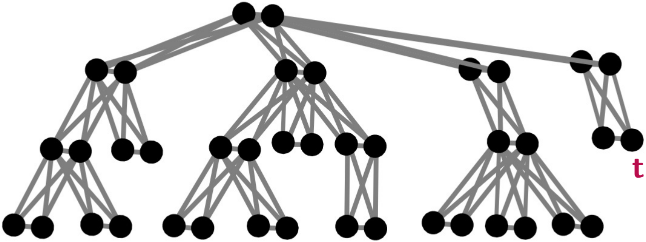
Modern trends in graph theory



tree

How can we design **high-dimensional** expanders?
How can we determine and utilize **structural** properties?

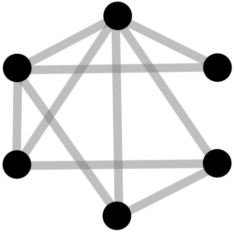
Modern trends in graph theory



small **tree-width**

How can we design **high-dimensional** expanders?
How can we determine and utilize **structural** properties?

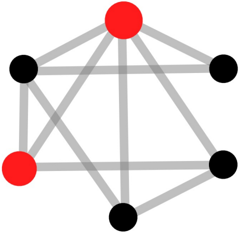
Modern trends in graph theory



query complexity

How can we design **high-dimensional** expanders?
How can we determine and utilize **structural** properties?
Can **quantum computing** speed up classical graph problems?

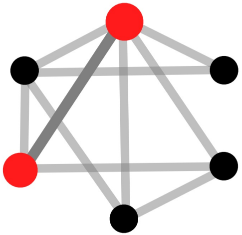
Modern trends in graph theory



query complexity

How can we design **high-dimensional** expanders?
How can we determine and utilize **structural** properties?
Can **quantum computing** speed up classical graph problems?

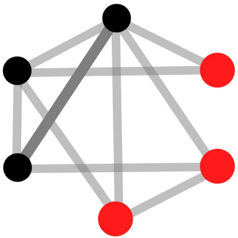
Modern trends in graph theory



query complexity

How can we design **high-dimensional** expanders?
How can we determine and utilize **structural** properties?
Can **quantum computing** speed up classical graph problems?

Modern trends in graph theory



query complexity

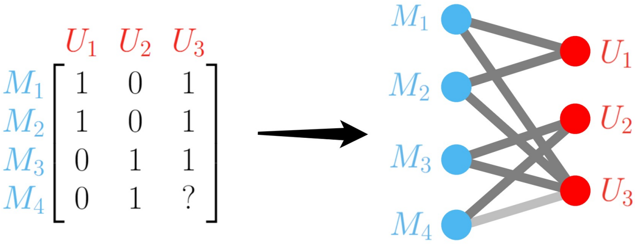
How can we design **high-dimensional** expanders?
How can we determine and utilize **structural** properties?
Can **quantum computing** speed up classical graph problems?

Modern trends in graph theory

$$\begin{array}{c} \\ M_1 \\ M_2 \\ M_3 \\ M_4 \end{array} \begin{array}{c} \begin{array}{ccc} U_1 & U_2 & U_3 \end{array} \\ \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & ? \end{bmatrix} \end{array}$$

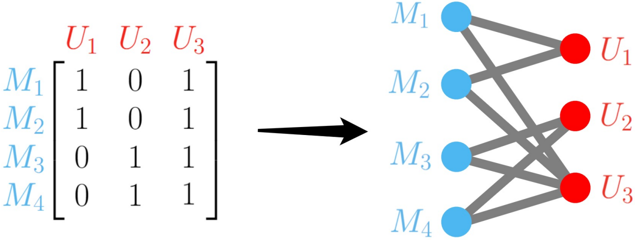
How can we design **high-dimensional** expanders?
How can we determine and utilize **structural** properties?
Can **quantum computing** speed up classical graph problems?

Modern trends in graph theory



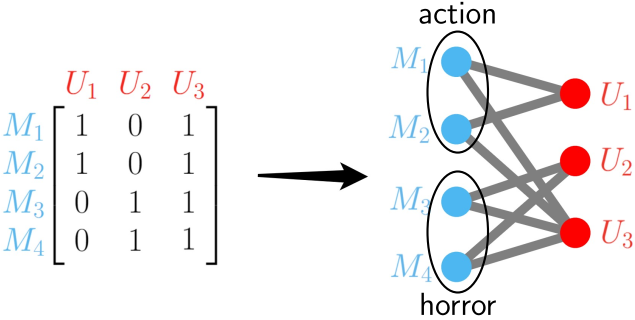
How can we design **high-dimensional** expanders?
How can we determine and utilize **structural** properties?
Can **quantum computing** speed up classical graph problems?
What if we care about the rank of **matrices**?

Modern trends in graph theory



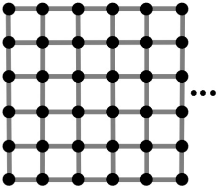
How can we design **high-dimensional** expanders?
How can we determine and utilize **structural** properties?
Can **quantum computing** speed up classical graph problems?
What if we care about the rank of **matrices**?

Modern trends in graph theory



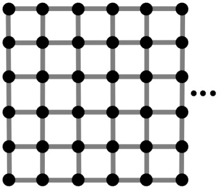
How can we design **high-dimensional** expanders?
How can we determine and utilize **structural** properties?
Can **quantum computing** speed up classical graph problems?
What if we care about the rank of **matrices**?

A graph-based model of quantum computing



(Raussendorf & Briegel 2001)

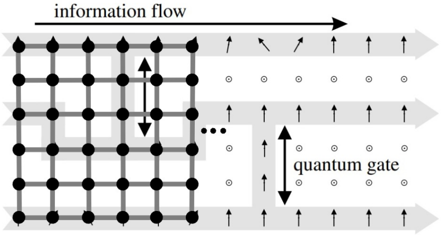
A graph-based model of quantum computing



(Raussendorf & Briegel 2001)

1) Given an n-vertex graph G, prepare an n-qubit state $|G\rangle$.

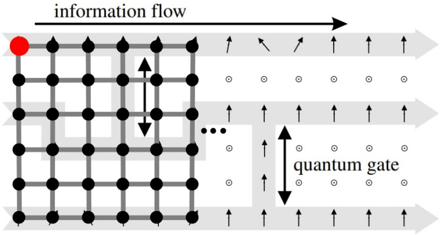
A graph-based model of quantum computing



(Raussendorf & Briegel 2001)

1) Given an n-vertex graph G, prepare an n-qubit state $|G\rangle$.
2) Use classical computing to determine qubits to measure.

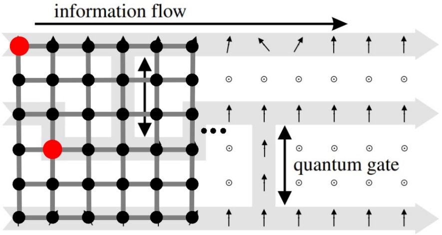
A graph-based model of quantum computing



(Raussendorf & Briegel 2001)

1) Given an n-vertex graph G, prepare an n-qubit state $|G\rangle$.
2) Use classical computing to determine qubits to measure.

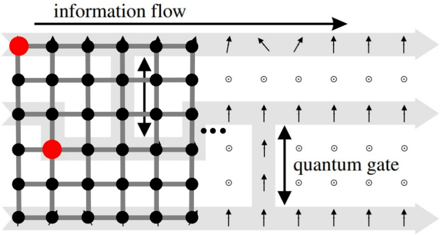
A graph-based model of quantum computing



(Raussendorf & Briegel 2001)

1) Given an n-vertex graph G, prepare an n-qubit state $|G\rangle$.
2) Use classical computing to determine qubits to measure.

A graph-based model of quantum computing

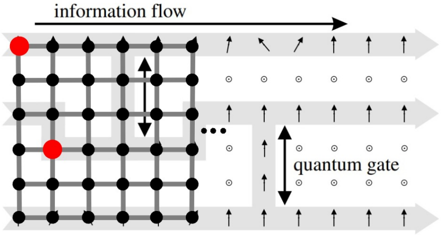


(Raussendorf & Briegel 2001)

1) Given an n-vertex graph G, prepare an n-qubit state $|G\rangle$.
2) Use classical computing to determine qubits to measure.

Theorem: Roughly equivalent to quantum gate model.

A graph-based model of quantum computing



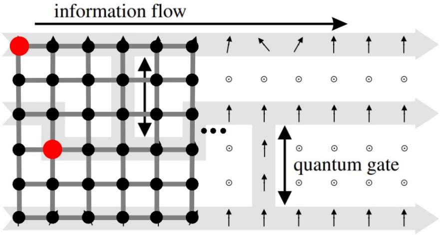
(Raussendorf & Briegel 2001)

1) Given an n-vertex graph G, prepare an n-qubit state $|G\rangle$.
2) Use classical computing to determine qubits to measure.

Theorem: Roughly equivalent to quantum gate model.

Q: What resources are necessary for quantum computing?

A graph-based model of quantum computing



(Raussendorf & Briegel 2001)

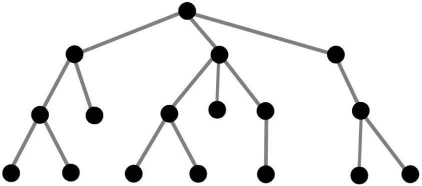
1) Given an n-vertex graph G, prepare an n-qubit state $|G\rangle$.
2) Use classical computing to determine qubits to measure.

Theorem: Roughly equivalent to quantum gate model.

Q: What resources are necessary for quantum computing?
Conjecture (Geelen)**: All of them;** no proper fragment of quantum computing is stronger than classical computing.

A graph-based model of quantum computing

Theorem: True when G has small **rank-width**.



(Van den Nest, Dür, Vidal, & Briegel 2007)

1) Given an n-vertex graph G, prepare an n-qubit state $|G\rangle$.
2) Use classical computing to determine qubits to measure.

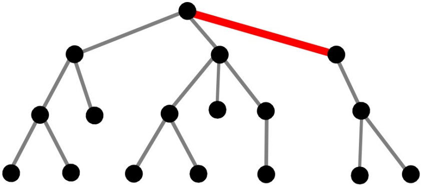
Theorem: Roughly equivalent to quantum gate model.

Q: What resources are necessary for quantum computing?

Conjecture (Geelen)**: All of them;** no proper fragment of quantum computing is stronger than classical computing.

A graph-based model of quantum computing

Theorem: True when G has small **rank-width**.



(Van den Nest, Dür, Vidal, & Briegel 2007)

1) Given an n-vertex graph G, prepare an n-qubit state $|G\rangle$.
2) Use classical computing to determine qubits to measure.

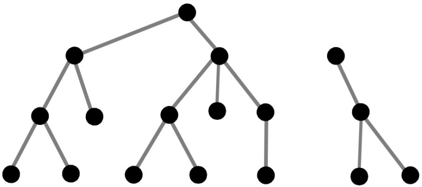
Theorem: Roughly equivalent to quantum gate model.

Q: What resources are necessary for quantum computing?

Conjecture (Geelen)**: All of them;** no proper fragment of quantum computing is stronger than classical computing.

A graph-based model of quantum computing

Theorem: True when G has small **rank-width**.



(Van den Nest, Dür, Vidal, & Briegel 2007)

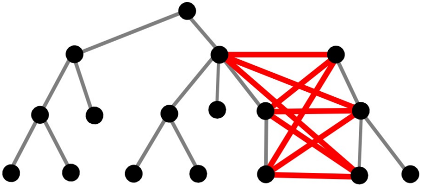
1) Given an n-vertex graph G, prepare an n-qubit state $|G\rangle$.
2) Use classical computing to determine qubits to measure.

Theorem: Roughly equivalent to quantum gate model.

Q: What resources are necessary for quantum computing?
Conjecture (Geelen)**: All of them;** no proper fragment of quantum computing is stronger than classical computing.

A graph-based model of quantum computing

Theorem: True when G has small **rank-width**.



(Van den Nest, Dür, Vidal, & Briegel 2007)

1) Given an n-vertex graph G, prepare an n-qubit state $|G\rangle$.
2) Use classical computing to determine qubits to measure.

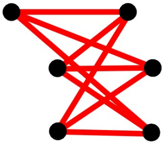
Theorem: Roughly equivalent to quantum gate model.

Q: What resources are necessary for quantum computing?
Conjecture (Geelen)**: All of them;** no proper fragment of quantum computing is stronger than classical computing.

A graph-based model of quantum computing

Theorem: True when G has small **rank-width**.

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$



(Van den Nest, Dür, Vidal, & Briegel 2007)

1) Given an n-vertex graph G, prepare an n-qubit state $|G\rangle$.
2) Use classical computing to determine qubits to measure.

Theorem: Roughly equivalent to quantum gate model.

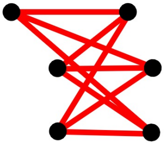
Q: What resources are necessary for quantum computing?

Conjecture (Geelen)**: All of them;** no proper fragment of quantum computing is stronger than classical computing.

A graph-based model of quantum computing

Theorem: True when G has small **rank-width**.

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$



Rank captures the level of entanglement.

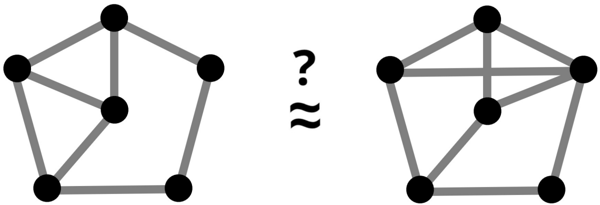
(Van den Nest, Dür, Vidal, & Briegel 2007)

1) Given an n-vertex graph G, prepare an n-qubit state $|G\rangle$.
2) Use classical computing to determine qubits to measure.

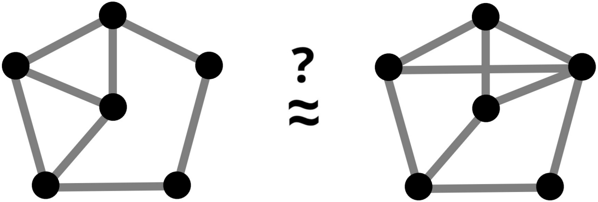
Theorem: Roughly equivalent to quantum gate model.

Q: What resources are necessary for quantum computing?
Conjecture (Geelen)**: All of them;** no proper fragment of quantum computing is stronger than classical computing.

When are $|G\rangle$ and $|G'\rangle$ equivalent?



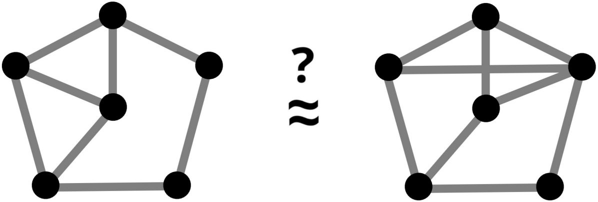
When are $|G\rangle$ and $|G'\rangle$ equivalent?



Two states are **LC-equivalent** if one can be transformed into the other by applying unitaries that:

- act on one qubit, and

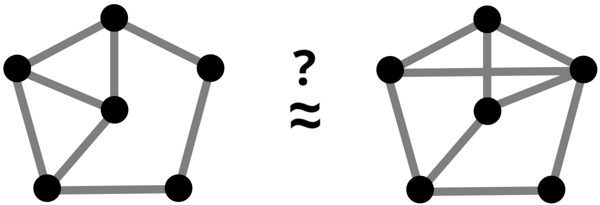
When are $|G\rangle$ and $|G'\rangle$ equivalent?



Two states are **LC-equivalent** if one can be transformed into the other by applying unitaries that:

- act on one qubit, and
- normalize the Pauli group, i.e., satisfy $U\mathcal{P}_1 U^\intercal = \mathcal{P}_1$.

When are $|G\rangle$ and $|G'\rangle$ equivalent?

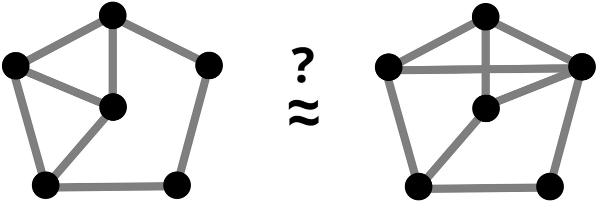


Two states are **LC-equivalent** if one can be transformed into the other by applying unitaries that:

- act on one qubit, and
- normalize the Pauli group, i.e., satisfy $U\mathcal{P}_1 U^\intercal = \mathcal{P}_1$.

Theorem (Van den Nest, Dehaene, & De Moor 2004)**:**
$|G\rangle$ and $|G'\rangle$ **LC-equivalent** $\Leftrightarrow$ G and G' **locally equivalent**

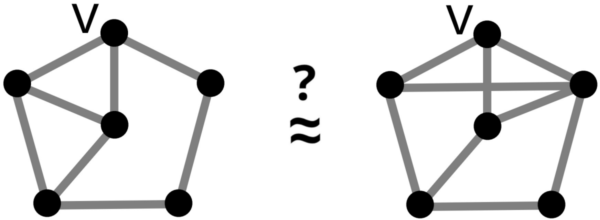
When are $|G\rangle$ and $|G'\rangle$ equivalent?



Two graphs are **locally-equivalent** if one can be transformed into the other by applying **local complementations**. For this:

Theorem (Van den Nest, Dehaene, & De Moor 2004)**:**
$|G\rangle$ and $|G'\rangle$ **LC-equivalent** $\Leftrightarrow$ G and G' **locally equivalent**

When are $|G\rangle$ and $|G'\rangle$ equivalent?

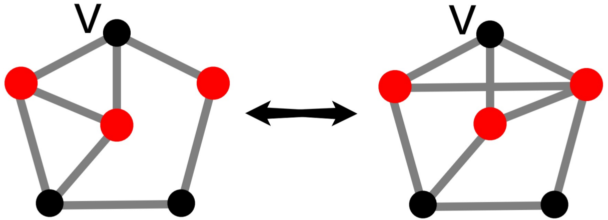


Two graphs are **locally-equivalent** if one can be transformed into the other by applying **local complementations**. For this:

- select a vertex **v**, then

Theorem (Van den Nest, Dehaene, & De Moor 2004)**:**
$|G\rangle$ and $|G'\rangle$ **LC-equivalent** $\Leftrightarrow$ G and G' **locally equivalent**

When are $|G\rangle$ and $|G'\rangle$ equivalent?

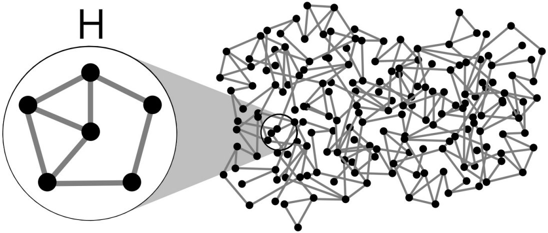


Two graphs are **locally-equivalent** if one can be transformed into the other by applying **local complementations**. For this:

- select a vertex **v**, then
- switch adjacencies within the **neighborhood** of v.

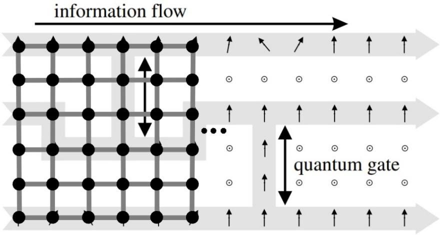
Theorem (Van den Nest, Dehaene, & De Moor 2004)**:**
$|G\rangle$ and $|G'\rangle$ **LC-equivalent** $\Leftrightarrow$ G and G' **locally equivalent**

Forbidding a vertex-minor



A graph G forbids a graph H as a **vertex-minor** if no graph that is locally equivalent to G contains a copy of H.

Forbidding a vertex-minor

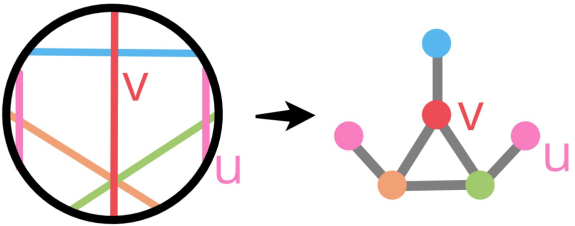


A graph G forbids a graph H as a **vertex-minor** if no graph that is locally equivalent to G contains a copy of H.

Conjecture (Geelen)**:** If G forbids a vertex-minor H, then

1) we can **classically simulate** quantum computing on $|G\rangle$,

Forbidding a vertex-minor

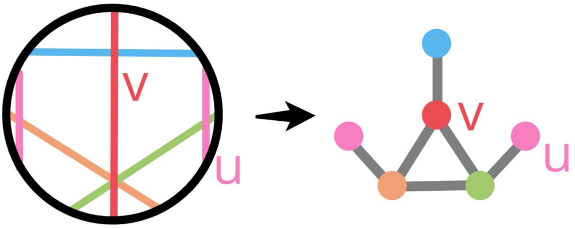


A graph G forbids a graph H as a **vertex-minor** if no graph that is locally equivalent to G contains a copy of H.

Conjecture (Geelen)**:** If G forbids a vertex-minor H, then

1) we can **classically simulate** quantum computing on $|G\rangle$,
2) G decomposes into parts that are almost **circle graphs**.

Forbidding a vertex-minor

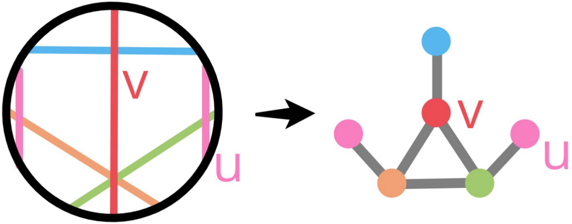


A graph G forbids a graph H as a **vertex-minor** if no graph that is locally equivalent to G contains a copy of H.

Conjecture (Geelen)**:** If G forbids a vertex-minor H, then

1) we can **classically simulate** quantum computing on $|G\rangle$,
2) G decomposes into parts that are almost **circle graphs**.

Forbidding a vertex-minor



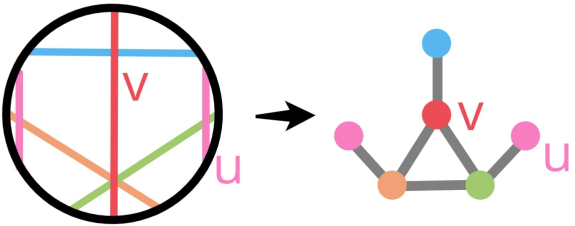
A graph G forbids a graph H as a **vertex-minor** if no graph that is locally equivalent to G contains a copy of H.

Conjecture (Geelen)**:** If G forbids a vertex-minor H, then

1) we can **classically simulate** quantum computing on $|G\rangle$,
2) G decomposes into parts that are almost **circle graphs**.

The grid theorem for vertex-minors
J Geelen, O Kwon, R McCarty, P Wollan
Journal of Combinatorial Theory, Series B

Forbidding a vertex-minor



A graph G forbids a graph H as a **vertex-minor** if no graph that is locally equivalent to G contains a copy of H.

Conjecture (Geelen)**:** If G forbids a vertex-minor H, then

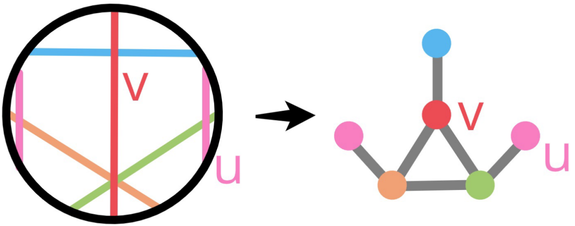
1) we can **classically simulate** quantum computing on $|G\rangle$,
2) G decomposes into parts that are almost **circle graphs**.

Local Structure for Vertex-Minors
R McCarty
University of Waterloo

The grid theorem for vertex-minors
J Geelen, O Kwon, R McCarty, P Wollan
Journal of Combinatorial Theory, Series B

Decomposing a signed graph into rooted circuits
R McCarty
Advances in Combinatorics.

Forbidding a vertex-minor



A graph G forbids a graph H as a **vertex-minor** if no graph that is locally equivalent to G contains a copy of H.

Conjecture (Geelen)**:** If G forbids a vertex-minor H, then

1) we can **classically simulate** quantum computing on $|G\rangle$,
2) G decomposes into parts that are almost **circle graphs**.

Local Structure for Vertex-Minors
R McCarty
University of Waterloo

The grid theorem for vertex-minors
J Geelen, O Kwon, R McCarty, P Wollan
Journal of Combinatorial Theory, Series B

Decomposing a signed graph into rooted circuits
R McCarty
Advances in Combinatorics.

Obstructions for bounded shrub-depth and rank-depth
O Kwon, R McCarty, S Oum, P Wollan
Journal of Combinatorial Theory, Series B 149, 76-91

Figures

- https://www.arcgis.com/apps/mapviewer/index.html?layers=d4090758322c4d32a4cd002ffaa0aa12
- https://simons.berkeley.edu/programs/summer-cluster-error-correcting-codes-high-dimensional-expansion
- https://www.ajc.com/lifestyles/flashback-photos-spaghetti-junction-through-the-years/YcgLW4PA0bGnjBLOEv8JbO/
- https://www.cbo.gov/publication/56990
- https://stock.adobe.com/images/diamond-crystal-structure-model-3d/44740015
- https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.86.5188

Thank you!