Quantum walks through generalized graph composition

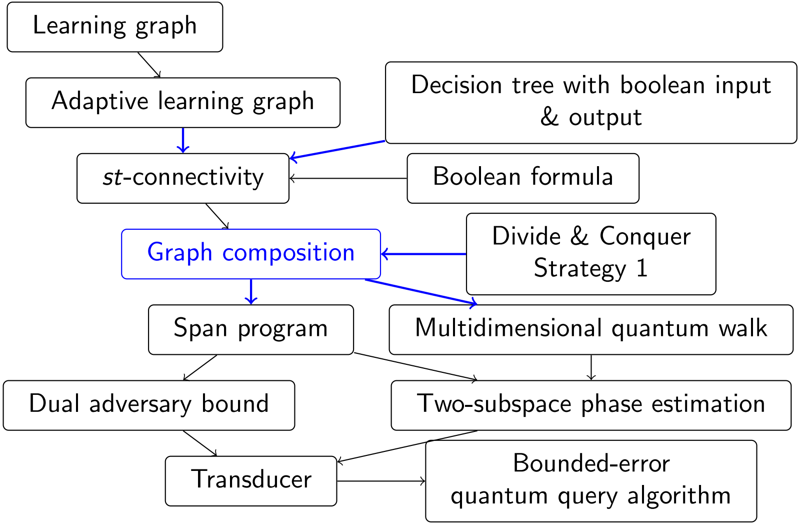
Arjan Cornelissen[1]

[1]Simons Institute, University of California, Berkeley, California

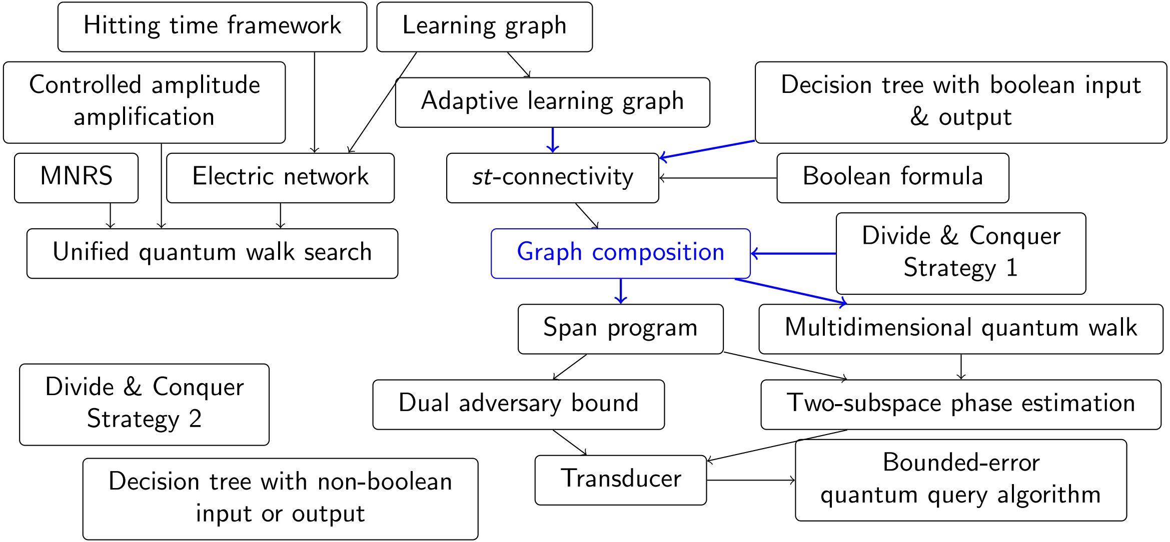
September 30th, 2025

Overview of quantum algorithmic frameworks

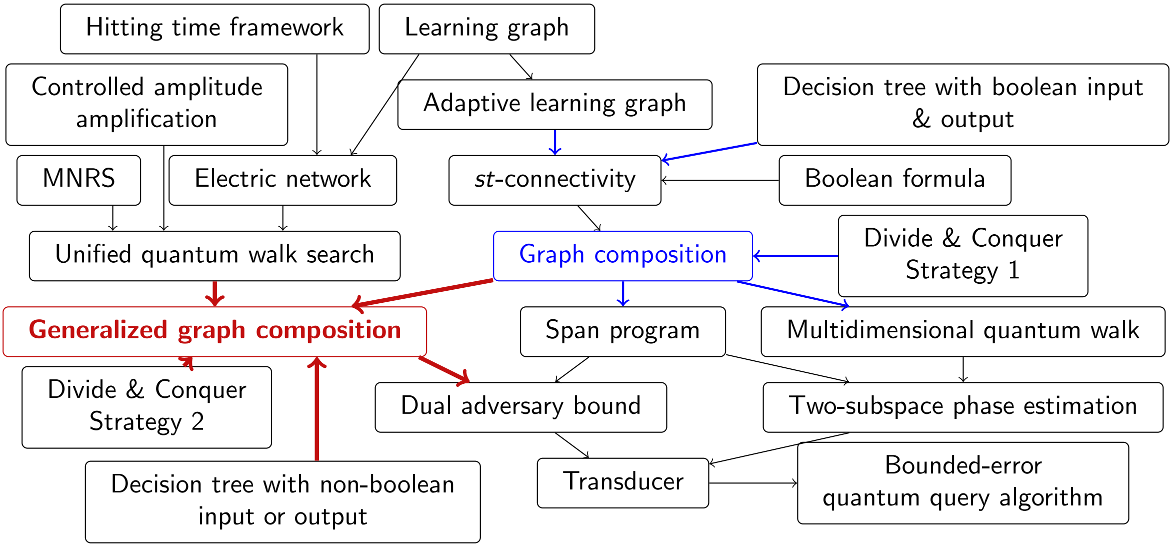
Overview of quantum algorithmic frameworks



Overview of quantum algorithmic frameworks



Overview of quantum algorithmic frameworks



State-conversion problems

State-conversion problems

State-conversion problem: [LMR+11]
$P = \{(|\sigma_x\rangle, |\tau_x\rangle, O_x)\}_{x \in \mathcal{D}}$:

1. *Input states:* $|\sigma_x\rangle \in \mathcal{V}_I$.
2. *Output states:* $|\tau_x\rangle \in \mathcal{V}_O$.
3. *Oracle:* $O_x \in \mathcal{L}(\mathcal{H})$ unitary.

Goal: $\forall x \in \mathcal{D}, |\sigma_x\rangle \overset{\mathcal{A}(O_x)}{\rightsquigarrow} |\tau_x\rangle$.

State-conversion problems

State-conversion problem: [LMR+11]

$P = \{(|\sigma_x\rangle, |\tau_x\rangle, O_x)\}_{x \in \mathcal{D}}$:

1. *Input states:* $|\sigma_x\rangle \in \mathcal{V}_I$.
2. *Output states:* $|\tau_x\rangle \in \mathcal{V}_O$.
3. *Oracle:* $O_x \in \mathcal{L}(\mathcal{H})$ unitary.

Goal: $\forall x \in \mathcal{D}, |\sigma_x\rangle \overset{\mathcal{A}(O_x)}{\rightsquigarrow} |\tau_x\rangle$.

Adversary bound: $\mathrm{ADV}(P) :=$

$$
\begin{aligned}
\min \quad & \max_{x \in \mathcal{D}} \left\| |w_x\rangle \right\|^2, \\
\text{s.t.} \quad & \langle w_x| \left(I_{\mathcal{H}\otimes\mathcal{W}} - (O_x^\dagger O_y) \otimes I_{\mathcal{W}}\right) |w_y\rangle \\
& = \langle \sigma_x|\sigma_y\rangle - \langle \tau_x|\tau_y\rangle, \forall x, y \in \mathcal{D}, \\
& |w_x\rangle \in \mathcal{H} \otimes \mathcal{W}, \forall x \in \mathcal{D}, \\
& \mathcal{W} \text{ Hilbert space.}
\end{aligned}
$$

State-conversion problems

State-conversion problem: [LMR+11]
$P = \{(|\sigma_x\rangle, |\tau_x\rangle, O_x)\}_{x \in \mathcal{D}}$:

1. *Input states:* $|\sigma_x\rangle \in \mathcal{V}_I$.
2. *Output states:* $|\tau_x\rangle \in \mathcal{V}_O$.
3. *Oracle:* $O_x \in \mathcal{L}(\mathcal{H})$ unitary.

Goal: $\forall x \in \mathcal{D}, |\sigma_x\rangle \overset{\mathcal{A}(O_x)}{\rightsquigarrow} |\tau_x\rangle$.
Adversary bound: $\mathrm{ADV}(P) :=$

$$\begin{aligned} \min \quad & \max_{x \in \mathcal{D}} \| |w_x\rangle \|^2, \\ \text{s.t.} \quad & \langle w_x | (I_{\mathcal{H}\otimes\mathcal{W}} - (O_x^\dagger O_y) \otimes I_{\mathcal{W}}) | w_y \rangle \\ & = \langle \sigma_x | \sigma_y \rangle - \langle \tau_x | \tau_y \rangle, \forall x, y \in \mathcal{D}, \\ & |w_x\rangle \in \mathcal{H} \otimes \mathcal{W}, \forall x \in \mathcal{D}, \\ & \mathcal{W} \text{ Hilbert space.} \end{aligned}$$

Result: [BJY24] we can build $\mathcal{A}$ with

$$\|\mathcal{A}(O_x)|\sigma_x\rangle - |\tau_x\rangle\|^2 \in O\left(\frac{\mathrm{ADV}(P)}{K}\right),$$

with K queries to O_x.

State-conversion problems

State-conversion problem: [LMR+11]

$P = \{(|\sigma_x\rangle, |\tau_x\rangle, O_x)\}_{x \in \mathcal{D}}$:

1. *Input states:* $|\sigma_x\rangle \in \mathcal{V}_I$.
2. *Output states:* $|\tau_x\rangle \in \mathcal{V}_O$.
3. *Oracle:* $O_x \in \mathcal{L}(\mathcal{H})$ unitary.

Goal: $\forall x \in \mathcal{D}, |\sigma_x\rangle \overset{\mathcal{A}(O_x)}{\rightsquigarrow} |\tau_x\rangle$.

Adversary bound: $\mathrm{ADV}(P) :=$

$$\begin{aligned} \min \quad & \max_{x \in \mathcal{D}} \| |w_x\rangle \|^2, \\ \text{s.t.} \quad & \langle w_x | (I_{\mathcal{H} \otimes \mathcal{W}} - (O_x^\dagger O_y) \otimes I_{\mathcal{W}}) | w_y \rangle \\ & = \langle \sigma_x | \sigma_y \rangle - \langle \tau_x | \tau_y \rangle, \forall x, y \in \mathcal{D}, \\ & |w_x\rangle \in \mathcal{H} \otimes \mathcal{W}, \forall x \in \mathcal{D}, \\ & \mathcal{W} \text{ Hilbert space.} \end{aligned}$$

Result: [BJY24] we can build $\mathcal{A}$ with

$$\| \mathcal{A}(O_x) |\sigma_x\rangle - |\tau_x\rangle \|^2 \in O\left(\frac{\mathrm{ADV}(P)}{K}\right),$$

with K queries to O_x.

Special cases:

1. *Database update:*
   1. $|\sigma_x\rangle, |\tau_x\rangle$ comp. basis states.
2. *Function evaluation:* $f : \mathcal{D} \to \Sigma$
   1. $|\sigma_x\rangle = |\perp\rangle$, $\forall x \in \mathcal{D}$.
   2. $|\tau_x\rangle = |f(x)\rangle$, $\forall x \in \mathcal{D}$.

State-reflection problems

State-reflection problems

State-reflection problem:

$R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x)\}_{x \in \mathcal{D}}$.

1. *Reflection states:* $|\sigma_x^\pm\rangle \in \mathcal{V}$.
   - *Orthogonality:* $\langle\sigma_x^+|\sigma_x^-\rangle = 0$.

State-reflection problems

State-reflection problem:
$R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x)\}_{x \in \mathcal{D}}$.

1. *Reflection states:* $|\sigma_x^\pm\rangle \in \mathcal{V}$.
   - *Orthogonality:* $\langle \sigma_x^+ | \sigma_x^- \rangle = 0$.
2. *Oracle:* $O_x \in \mathcal{L}(\mathcal{H})$.
   - *Involution:* $O_x^2 = I$.

State-reflection problems

State-reflection problem:
$R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x)\}_{x \in \mathcal{D}}$.

1. *Reflection states:* $|\sigma_x^\pm\rangle \in \mathcal{V}$.
    - *Orthogonality:* $\langle\sigma_x^+|\sigma_x^-\rangle = 0$.
2. *Oracle:* $O_x \in \mathcal{L}(\mathcal{H})$.
    - *Involution:* $O_x^2 = I$.

Goal: $|\sigma_x^\pm\rangle \overset{\mathcal{A}(O_x)}{\rightsquigarrow} \pm|\sigma_x^\pm\rangle$.

State-reflection problems

State-reflection problem:
$R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x)\}_{x \in \mathcal{D}}$.

1. *Reflection states:* $|\sigma_x^\pm\rangle \in \mathcal{V}$.
   - *Orthogonality:* $\langle \sigma_x^+ | \sigma_x^- \rangle = 0$.
2. *Oracle:* $O_x \in \mathcal{L}(\mathcal{H})$.
   - *Involution:* $O_x^2 = I$.

Goal: $|\sigma_x^\pm\rangle \overset{\mathcal{A}(O_x)}{\rightsquigarrow} \pm |\sigma_x^\pm\rangle$.
Adversary bound: $\mathrm{ADV}(R) :=$

$$\begin{aligned}
\min \quad & \max\{\max_{x \in \mathcal{D}} \left\| |w_x^+\rangle \right\|^2, \max_{x \in \mathcal{D}} \left\| |w_x^-\rangle \right\|^2\}, \\
\text{s.t.} \quad & \left\langle \sigma_x^+ \middle| \sigma_y^- \right\rangle = \left\langle w_x^+ \middle| w_y^- \right\rangle, \forall x, y \in \mathcal{D}, \\
& (O_x \otimes I_{\mathcal{W}}) |w_x^\pm\rangle = \pm |w_x^\pm\rangle, \forall x \in \mathcal{D}, \\
& |w_x^\pm\rangle \in \mathcal{H} \otimes \mathcal{W}, \forall x \in \mathcal{D}, \\
& \mathcal{W} \text{ Hilbert space.}
\end{aligned}$$

State-reflection problems

State-reflection problem:
$R = \{(\left|\sigma_x^+\right\rangle, \left|\sigma_x^-\right\rangle, O_x)\}_{x \in \mathcal{D}}$.

1. *Reflection states:* $\left|\sigma_x^\pm\right\rangle \in \mathcal{V}$.
   - *Orthogonality:* $\langle\sigma_x^+|\sigma_x^-\rangle = 0$.
2. *Oracle:* $O_x \in \mathcal{L}(\mathcal{H})$.
   - *Involution:* $O_x^2 = I$.

Goal: $\left|\sigma_x^\pm\right\rangle \overset{\mathcal{A}(O_x)}{\rightsquigarrow} \pm\left|\sigma_x^\pm\right\rangle$.
Adversary bound: $\mathrm{ADV}(R) :=$

$$\begin{aligned} \min \quad & \max\{\max_{x \in \mathcal{D}} \left\|\left|w_x^+\right\rangle\right\|^2, \max_{x \in \mathcal{D}} \left\|\left|w_x^-\right\rangle\right\|^2\}, \\ \text{s.t.} \quad & \left\langle\sigma_x^+\middle|\sigma_y^-\right\rangle = \left\langle w_x^+\middle|w_y^-\right\rangle, \forall x, y \in \mathcal{D}, \\ & (O_x \otimes I_{\mathcal{W}})\left|w_x^\pm\right\rangle = \pm\left|w_x^\pm\right\rangle, \forall x \in \mathcal{D}, \\ & \left|w_x^\pm\right\rangle \in \mathcal{H} \otimes \mathcal{W}, \forall x \in \mathcal{D}, \\ & \mathcal{W} \text{ Hilbert space.} \end{aligned}$$

Theorem:

1. $P = \{(\left|\sigma_x\right\rangle, \left|\tau_x\right\rangle, O_x)\}_{x \in \mathcal{D}}$ state-conversion problem.
2. Define $R = \{(\left|\sigma_x^+\right\rangle, \left|\sigma_x^-\right\rangle, O_x')\}_{x \in \mathcal{D}}$, with
   1. $\left|\sigma_x^\pm\right\rangle = \frac{\left|\sigma_x\right\rangle \oplus \pm\left|\tau_x\right\rangle}{\sqrt{2}}$
   2. $O_x' = (X \otimes I_{\mathcal{H}})(O_x \oplus O_x^\dagger)$
3. Then: $R \Leftrightarrow P$ (identical feasible regions).

State-reflection problems

State-reflection problem:
$R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x)\}_{x \in \mathcal{D}}$.

1. *Reflection states:* $|\sigma_x^\pm\rangle \in \mathcal{V}$.
   - *Orthogonality:* $\langle\sigma_x^+|\sigma_x^-\rangle = 0$.
2. *Oracle:* $O_x \in \mathcal{L}(\mathcal{H})$.
   - *Involution:* $O_x^2 = I$.

Goal: $|\sigma_x^\pm\rangle \overset{\mathcal{A}(O_x)}{\rightsquigarrow} \pm|\sigma_x^\pm\rangle$.

Adversary bound: $\mathrm{ADV}(R) :=$

$$\begin{aligned} \min \quad & \max\{\max_{x \in \mathcal{D}} \||w_x^+\rangle\|^2, \max_{x \in \mathcal{D}} \||w_x^-\rangle\|^2\}, \\ \text{s.t.} \quad & \langle\sigma_x^+|\sigma_y^-\rangle = \langle w_x^+|w_y^-\rangle, \forall x, y \in \mathcal{D}, \\ & (O_x \otimes I_{\mathcal{W}})|w_x^\pm\rangle = \pm|w_x^\pm\rangle, \forall x \in \mathcal{D}, \\ & |w_x^\pm\rangle \in \mathcal{H} \otimes \mathcal{W}, \forall x \in \mathcal{D}, \\ & \mathcal{W} \text{ Hilbert space.} \end{aligned}$$

Theorem:

1. $P = \{(|\sigma_x\rangle, |\tau_x\rangle, O_x)\}_{x \in \mathcal{D}}$ state-conversion problem.
2. Define $R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x')\}_{x \in \mathcal{D}}$, with
   1. $|\sigma_x^\pm\rangle = \frac{|\sigma_x\rangle \oplus \pm|\tau_x\rangle}{\sqrt{2}}$
   2. $O_x' = (X \otimes I_{\mathcal{H}})(O_x \oplus O_x^\dagger)$
3. Then: $R \Leftrightarrow P$ (identical feasible regions).

Properties:

State-reflection problems

State-reflection problem:
$R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x)\}_{x \in \mathcal{D}}$.

1. *Reflection states:* $|\sigma_x^\pm\rangle \in \mathcal{V}$.
   - *Orthogonality:* $\langle \sigma_x^+ | \sigma_x^- \rangle = 0$.
2. *Oracle:* $O_x \in \mathcal{L}(\mathcal{H})$.
   - *Involution:* $O_x^2 = I$.

Goal: $|\sigma_x^\pm\rangle \overset{\mathcal{A}(O_x)}{\rightsquigarrow} \pm |\sigma_x^\pm\rangle$.

Adversary bound: $\mathrm{ADV}(R) :=$

$$\begin{aligned} \min \quad & \max\{\max_{x \in \mathcal{D}} \| |w_x^+\rangle \|^2, \max_{x \in \mathcal{D}} \| |w_x^-\rangle \|^2\}, \\ \text{s.t.} \quad & \langle \sigma_x^+ | \sigma_y^- \rangle = \langle w_x^+ | w_y^- \rangle, \forall x, y \in \mathcal{D}, \\ & (O_x \otimes I_{\mathcal{W}}) |w_x^\pm\rangle = \pm |w_x^\pm\rangle, \forall x \in \mathcal{D}, \\ & |w_x^\pm\rangle \in \mathcal{H} \otimes \mathcal{W}, \forall x \in \mathcal{D}, \\ & \mathcal{W} \text{ Hilbert space.} \end{aligned}$$

Theorem:

1. $P = \{(|\sigma_x\rangle, |\tau_x\rangle, O_x)\}_{x \in \mathcal{D}}$ state-conversion problem.
2. Define $R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x')\}_{x \in \mathcal{D}}$, with
   1. $|\sigma_x^\pm\rangle = \frac{|\sigma_x\rangle \oplus \pm |\tau_x\rangle}{\sqrt{2}}$
   2. $O_x' = (X \otimes I_{\mathcal{H}})(O_x \oplus O_x^\dagger)$
3. Then: $R \Leftrightarrow P$ (identical feasible regions).

Properties:

1. *Positive/negative witness:* $|w_x^\pm\rangle$.

State-reflection problems

State-reflection problem:
$R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x)\}_{x \in \mathcal{D}}$.

1. *Reflection states:* $|\sigma_x^\pm\rangle \in \mathcal{V}$.
   - *Orthogonality:* $\langle \sigma_x^+ | \sigma_x^- \rangle = 0$.
2. *Oracle:* $O_x \in \mathcal{L}(\mathcal{H})$.
   - *Involution:* $O_x^2 = I$.

Goal: $|\sigma_x^\pm\rangle \overset{\mathcal{A}(O_x)}{\rightsquigarrow} \pm |\sigma_x^\pm\rangle$.

Adversary bound: $\mathrm{ADV}(R) :=$

$$\begin{array}{ll} \min & \max\{\max_{x \in \mathcal{D}} \left\| |w_x^+\rangle \right\|^2, \max_{x \in \mathcal{D}} \left\| |w_x^-\rangle \right\|^2\}, \\ \text{s.t.} & \langle \sigma_x^+ | \sigma_y^- \rangle = \langle w_x^+ | w_y^- \rangle, \forall x, y \in \mathcal{D}, \\ & (O_x \otimes I_{\mathcal{W}}) |w_x^\pm\rangle = \pm |w_x^\pm\rangle, \forall x \in \mathcal{D}, \\ & |w_x^\pm\rangle \in \mathcal{H} \otimes \mathcal{W}, \forall x \in \mathcal{D}, \\ & \mathcal{W} \text{ Hilbert space.} \end{array}$$

Theorem:

1. $P = \{(|\sigma_x\rangle, |\tau_x\rangle, O_x)\}_{x \in \mathcal{D}}$ state-conversion problem.
2. Define $R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x')\}_{x \in \mathcal{D}}$, with
   1. $|\sigma_x^\pm\rangle = \frac{|\sigma_x\rangle \oplus \pm |\tau_x\rangle}{\sqrt{2}}$
   2. $O_x' = (X \otimes I_{\mathcal{H}})(O_x \oplus O_x^\dagger)$
3. Then: $R \Leftrightarrow P$ (identical feasible regions).

Properties:

1. *Positive/negative witness:* $|w_x^\pm\rangle$.
2. Rescaling with constant factors.

State-reflection problems

State-reflection problem:

$R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x)\}_{x \in \mathcal{D}}$.

1. *Reflection states:* $|\sigma_x^\pm\rangle \in \mathcal{V}$.
   - *Orthogonality:* $\langle \sigma_x^+ | \sigma_x^- \rangle = 0$.
2. *Oracle:* $O_x \in \mathcal{L}(\mathcal{H})$.
   - *Involution:* $O_x^2 = I$.

Goal: $|\sigma_x^\pm\rangle \overset{\mathcal{A}(O_x)}{\rightsquigarrow} \pm |\sigma_x^\pm\rangle$.

Adversary bound: $\mathrm{ADV}(R) :=$

$$\begin{aligned} \min \quad & \max\{\max_{x \in \mathcal{D}} \left\| |w_x^+\rangle \right\|^2, \max_{x \in \mathcal{D}} \left\| |w_x^-\rangle \right\|^2\}, \\ \text{s.t.} \quad & \langle \sigma_x^+ | \sigma_y^- \rangle = \langle w_x^+ | w_y^- \rangle, \forall x, y \in \mathcal{D}, \\ & (O_x \otimes I_{\mathcal{W}}) |w_x^\pm\rangle = \pm |w_x^\pm\rangle, \forall x \in \mathcal{D}, \\ & |w_x^\pm\rangle \in \mathcal{H} \otimes \mathcal{W}, \forall x \in \mathcal{D}, \\ & \mathcal{W} \text{ Hilbert space.} \end{aligned}$$

Theorem:

1. $P = \{(|\sigma_x\rangle, |\tau_x\rangle, O_x)\}_{x \in \mathcal{D}}$ state-conversion problem.
2. Define $R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x')\}_{x \in \mathcal{D}}$, with
   1. $|\sigma_x^\pm\rangle = \frac{|\sigma_x\rangle \oplus \pm |\tau_x\rangle}{\sqrt{2}}$
   2. $O_x' = (X \otimes I_{\mathcal{H}})(O_x \oplus O_x^\dagger)$
3. Then: $R \Leftrightarrow P$ (identical feasible regions).

Properties:

1. *Positive/negative witness:* $|w_x^\pm\rangle$.
2. Rescaling with constant factors.
3. Only depends on inner products between pairs of $|\sigma_x^+\rangle$ and $|\sigma_y^-\rangle$.

Generalized graph composition (I/II)

State-reflection problem: $R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x)\}_{x \in \mathcal{D}}$.

Generalized graph composition (I/II)

State-reflection problem: $R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x)\}_{x \in \mathcal{D}}$.

Core idea: electric network interpretation:

1. V: vertices adjacent to a hyperedge.
2. $U_x := |\sigma_x^+\rangle \in \mathbb{C}^V$: potential function.
3. $\delta_x := |\sigma_x^-\rangle \in \mathbb{C}^V$: net-flow.

Generalized graph composition (I/II)

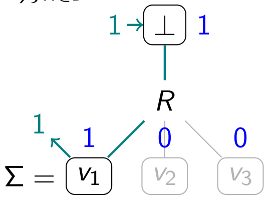
State-reflection problem: $R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x)\}_{x \in \mathcal{D}}$.

Core idea: electric network interpretation:

1. V: vertices adjacent to a hyperedge.
2. $U_x := |\sigma_x^+\rangle \in \mathbb{C}^V$: potential function.
3. $\delta_x := |\sigma_x^-\rangle \in \mathbb{C}^V$: net-flow.

Examples:

1. *Function evaluation:* $f : \mathcal{D} \to \Sigma$:
   1. $U_x = |\perp\rangle + |f(x)\rangle$.
   2. $\delta_x = |\perp\rangle - |f(x)\rangle$.
   3. $V = \{\perp\} \cup \Sigma$.



Function evaluation

Generalized graph composition (I/II)

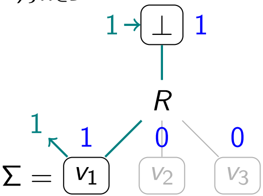
State-reflection problem: $R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x)\}_{x \in \mathcal{D}}$.

Core idea: electric network interpretation:

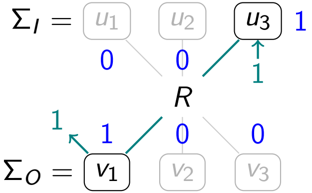
1. V: vertices adjacent to a hyperedge.
2. $U_x := |\sigma_x^+\rangle \in \mathbb{C}^V$: potential function.
3. $\delta_x := |\sigma_x^-\rangle \in \mathbb{C}^V$: net-flow.

Examples:

1. *Function evaluation:* $f : \mathcal{D} \to \Sigma$:
   1. $U_x = |\perp\rangle + |f(x)\rangle$.
   2. $\delta_x = |\perp\rangle - |f(x)\rangle$.
   3. $V = \{\perp\} \cup \Sigma$.
2. *Database update:* $\Sigma_I \to \Sigma_O$, $\sigma_x \mapsto \tau_x$:
   1. $U_x = |\sigma_x\rangle + |\tau_x\rangle$.
   2. $\delta_x = |\sigma_x\rangle - |\tau_x\rangle$.
   3. $V = \Sigma_I \cup \Sigma_O$.



Function evaluation



Database update

Generalized graph composition (I/II)

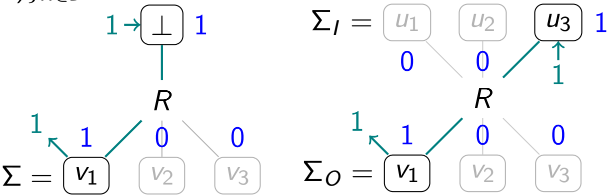
State-reflection problem: $R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x)\}_{x \in \mathcal{D}}$.

Core idea: electric network interpretation:

1. V: vertices adjacent to a hyperedge.
2. $U_x := |\sigma_x^+\rangle \in \mathbb{C}^V$: potential function.
3. $\delta_x := |\sigma_x^-\rangle \in \mathbb{C}^V$: net-flow.

Examples:

1. *Function evaluation:* $f : \mathcal{D} \to \Sigma$:
   1. $U_x = |\perp\rangle + |f(x)\rangle$.
   2. $\delta_x = |\perp\rangle - |f(x)\rangle$.
   3. $V = \{\perp\} \cup \Sigma$.
2. *Database update:* $\Sigma_I \to \Sigma_O$, $\sigma_x \mapsto \tau_x$:
   1. $U_x = |\sigma_x\rangle + |\tau_x\rangle$.
   2. $\delta_x = |\sigma_x\rangle - |\tau_x\rangle$.
   3. $V = \Sigma_I \cup \Sigma_O$.



Function evaluation *Database update*

On/off switch

3. *On/off switch:* $f : \mathcal{D} \to \{0, 1\}$,
   1. $U_x = \mathbb{1}_{f(x)=0} |s\rangle$.
   2. $\delta_x = \mathbb{1}_{f(x)=1}(|s\rangle - |t\rangle)$.
   3. $V = \{s, t\}$.

Generalized graph composition (I/II)

State-reflection problem: $R = \{(|\sigma_x^+\rangle, |\sigma_x^-\rangle, O_x)\}_{x \in \mathcal{D}}$.

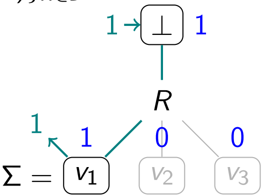
Core idea: electric network interpretation:

1. V: vertices adjacent to a hyperedge.
2. $U_x := |\sigma_x^+\rangle \in \mathbb{C}^V$: potential function.
3. $\delta_x := |\sigma_x^-\rangle \in \mathbb{C}^V$: net-flow.

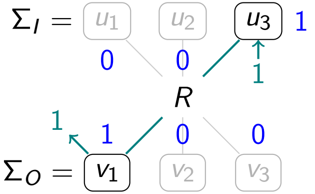
Examples:

1. *Function evaluation:* $f : \mathcal{D} \to \Sigma$:
   1. $U_x = |\perp\rangle + |f(x)\rangle$.
   2. $\delta_x = |\perp\rangle - |f(x)\rangle$.
   3. $V = \{\perp\} \cup \Sigma$.
2. *Database update:* $\Sigma_I \to \Sigma_O$,

   $\sigma_x \mapsto \tau_x$:
   1. $U_x = |\sigma_x\rangle + |\tau_x\rangle$.
   2. $\delta_x = |\sigma_x\rangle - |\tau_x\rangle$.
   3. $V = \Sigma_I \cup \Sigma_O$.



Function evaluation



Database update

On/off switch

3. *On/off switch:* $f : \mathcal{D} \to \{0, 1\}$,
   1. $U_x = \mathbb{1}_{f(x)=0} |s\rangle$.
   2. $\delta_x = \mathbb{1}_{f(x)=1}(|s\rangle - |t\rangle)$.
   3. $V = \{s, t\}$.

Theorem: Span program $\Leftrightarrow$ On/off switch.

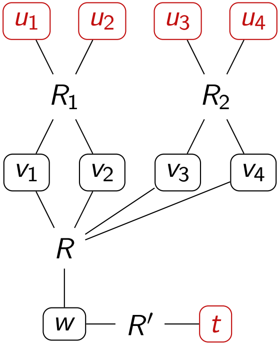
Generalized graph composition (II/II)

Core idea:

Generalized graph composition (II/II)

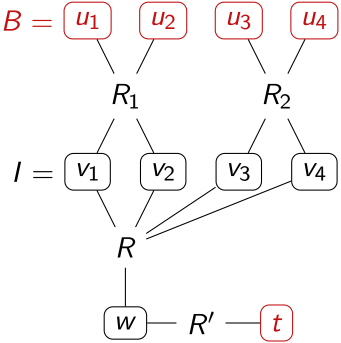
Core idea:

1. Hypergraph: $G = (V, E)$.



Core idea:

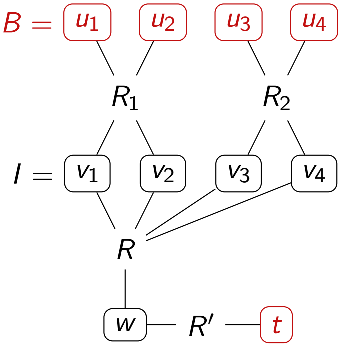
1. Hypergraph: $G = (V, E)$.
2. $V = B \sqcup I$ boundary and internal.



Generalized graph composition (II/II)

Core idea:

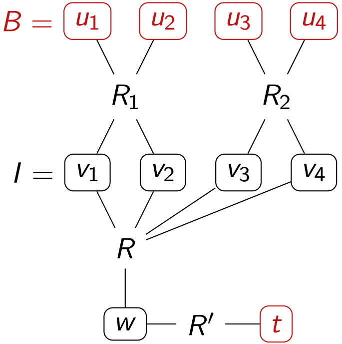
1. Hypergraph: $G = (V, E)$.
2. $V = B \sqcup I$ boundary and internal.
3. $\forall e \in E$, R^e state-reflection problem:
   1. such that net-flow vanishes on I.
   2. such that potential function is unique at every vertex.



Generalized graph composition (II/II)

Core idea:

1. Hypergraph: $G = (V, E)$.
2. $V = B \sqcup I$ boundary and internal.
3. $\forall e \in E$, R^e state-reflection problem:
   1. such that net-flow vanishes on I.
   2. such that potential function is unique at every vertex.
4. Defines U_x and δ_x on boundary vertices.
   $\Rightarrow R = \{(U_x, \delta_x, O_x)\}_{x \in \mathcal{D}}$.

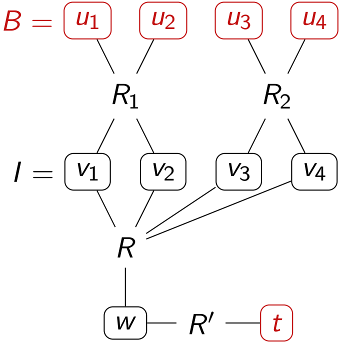


Generalized graph composition (II/II)

Core idea:

1. Hypergraph: $G = (V, E)$.
2. $V = B \sqcup I$ boundary and internal.
3. $\forall e \in E$, R^e state-reflection problem:
   1. such that net-flow vanishes on I.
   2. such that potential function is unique at every vertex.
4. Defines U_x and δ_x on boundary vertices.
   $\Rightarrow R = \{(U_x, \delta_x, O_x)\}_{x \in \mathcal{D}}$.

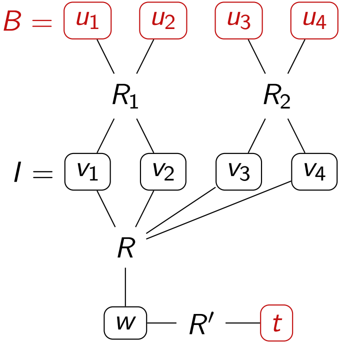
Theorem: $|w_x^{\pm}\rangle = \bigoplus_{e \in E} |(w^e)_x^{\pm}\rangle$ is a feas. sln.



Generalized graph composition (II/II)

Core idea:

1. Hypergraph: $G = (V, E)$.
2. $V = B \sqcup I$ boundary and internal.
3. $\forall e \in E$, R^e state-reflection problem:
   1. such that net-flow vanishes on I.
   2. such that potential function is unique at every vertex.
4. Defines U_x and δ_x on boundary vertices.
   $\Rightarrow R = \{(U_x, \delta_x, O_x)\}_{x \in \mathcal{D}}$.



Theorem: $|w_x^{\pm}\rangle = \bigoplus_{e \in E} |(w^e)_x^{\pm}\rangle$ is a feas. sln.

Proof: $\langle \sigma_x^+ | \sigma_y^- \rangle = U_x^T \delta_y = \sum_{v \in V} (U_x^e)_v \sum_{e \in N(v)} (\delta_x^e)_v = \sum_{e \in E} \sum_{v \in N(e)} (U_x^e)_v (\delta_x^e)_v$
$= \sum_{e \in E} (U_x^e)^T \delta_x^e = \sum_{e \in E} \langle (w_x^e)^+ | (w_y^e)^- \rangle = \langle w_x^+ | w_y^- \rangle$. $\square$

Example: finding first marked index

Example: finding first marked index

1. *Problem statement:*
   1. $\mathcal{D} = \{0,1\}^n \setminus \{0^n\}$.
   2. $f : \mathcal{D} \to [n]$,
   3. $f(x) = \min\{j \in [n] : x_j = 1\}$.

Example: finding first marked index

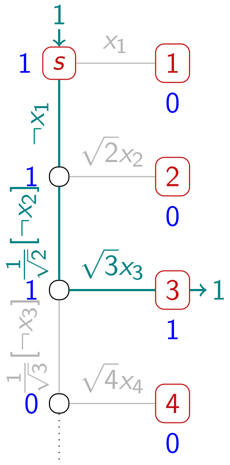
1. *Problem statement:*
   1. $\mathcal{D} = \{0,1\}^n \setminus \{0^n\}$.
   2. $f : \mathcal{D} \to [n]$,
   3. $f(x) = \min\{j \in [n] : x_j = 1\}$.
2. *Folklore:* $O(\sqrt{f(x)})$ queries suffice.

Example: finding first marked index

1. *Problem statement:*
    1. $\mathcal{D} = \{0,1\}^n \setminus \{0^n\}$.
    2. $f : \mathcal{D} \to [n]$,
    3. $f(x) = \min\{j \in [n] : x_j = 1\}$.
2. *Folklore:* $O(\sqrt{f(x)})$ queries suffice.
3. *Canonical on/off edge:* $\begin{tikzcd} \boxed{s} \arrow[r, dash, "\alpha x_j"] & \boxed{t} \end{tikzcd}$:
    1. $\||w_x^+\rangle\|^2 = \begin{cases} \alpha, & \text{if } x_j = 1, \\ 0, & \text{if } x_j = 0. \end{cases}, \quad \||w_x^-\rangle\|^2 = \begin{cases} 0, & \text{if } x_j = 1, \\ \frac{1}{\alpha}, & \text{if } x_j = 0. \end{cases}$

Example: finding first marked index

1. *Problem statement:*
   1. $\mathcal{D} = \{0,1\}^n \setminus \{0^n\}$.
   2. $f : \mathcal{D} \to [n]$,
   3. $f(x) = \min\{j \in [n] : x_j = 1\}$.
2. *Folklore:* $O(\sqrt{f(x)})$ queries suffice.
3. *Canonical on/off edge:* $\boxed{s} \overset{\alpha x_j}{\text{———}} \boxed{t}$:
   1. $$\||w_x^+\rangle\|^2 = \begin{cases} \alpha, & \text{if } x_j = 1, \\ 0, & \text{if } x_j = 0. \end{cases}, \qquad \||w_x^-\rangle\|^2 = \begin{cases} 0, & \text{if } x_j = 1, \\ \frac{1}{\alpha}, & \text{if } x_j = 0. \end{cases}$$
4. *Analysis:*
   1. $$\||w_x^+\rangle\|^2 = \sum_{e \in E} \||(w^e)_x^+\rangle\|^2 = \sum_{j=1}^{f(x)-1} \frac{1}{\sqrt{j}} + \sqrt{f(x)} \in O(\sqrt{f(x)}).$$
   2. $$\||w_x^-\rangle\|^2 = \sum_{e \in E} \||(w^e)_x^-\rangle\|^2 = \sum_{j=1}^{f(x)-1} \frac{1}{\sqrt{j}} + \sqrt{f(x)} \in O(\sqrt{f(x)}).$$



$x = 00101\ldots$

Application: Random walk search

Application: Random walk search

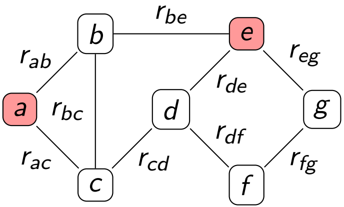
Walk search problem:

1. $G = (V, E)$: undirected graph.
2. $r : E \to \mathbb{R}_{>0}$: resistance function.
3. Finite set $\mathcal{D}$: domain.
4. $\forall x \in \mathcal{D}$, $M_x \subseteq V$: marked set.

Application: Random walk search

Walk search problem:

1. $G = (V, E)$: undirected graph.
2. $r : E \to \mathbb{R}_{>0}$: resistance function.
3. Finite set $\mathcal{D}$: domain.
4. $\forall x \in \mathcal{D}$, $M_x \subseteq V$: marked set.



$M_x = \{a, e\}$

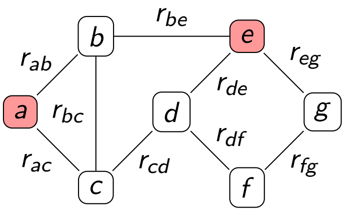
Application: Random walk search

Walk search problem:

1. $G = (V, E)$: undirected graph.
2. $r : E \to \mathbb{R}_{>0}$: resistance function.
3. Finite set $\mathcal{D}$: domain.
4. $\forall x \in \mathcal{D}$, $M_x \subseteq V$: marked set.

Goal:

1. *Detection:* Output whether $M_x \neq \varnothing$.
2. *Finding:* Output any $v \in M_x$, with $M_x \neq \varnothing$.



$M_x = \{a, e\}$

Application: Random walk search

Walk search problem:

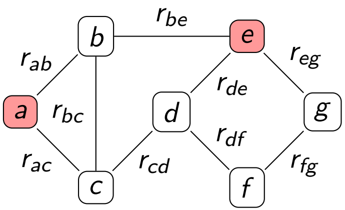
1. $G = (V, E)$: undirected graph.
2. $r : E \to \mathbb{R}_{>0}$: resistance function.
3. Finite set $\mathcal{D}$: domain.
4. $\forall x \in \mathcal{D}$, $M_x \subseteq V$: marked set.

Goal:

1. *Detection:* Output whether $M_x \neq \varnothing$.
2. *Finding:* Output any $v \in M_x$, with $M_x \neq \varnothing$.

Subroutines: $\forall x \in \mathcal{D}, v \in V$, $D_{v,x} \in \mathcal{S}$ *(data)*.

1. *Setup:* $v \mapsto D_{v,x}$ with S queries.
2. *Update:* $(vw, D_{v,x}) \mapsto D_{w,x}$, with U queries.
3. *Checking:* $(v, D_{v,x}) \mapsto [v \in M_x]$, with C queries.



$M_x = \{a, e\}$

Application: Random walk search

Walk search problem:

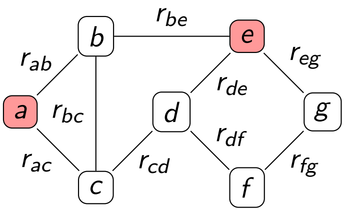
1. $G = (V, E)$: undirected graph.
2. $r : E \to \mathbb{R}_{>0}$: resistance function.
3. Finite set $\mathcal{D}$: domain.
4. $\forall x \in \mathcal{D}$, $M_x \subseteq V$: marked set.

Goal:

1. *Detection:* Output whether $M_x \neq \varnothing$.
2. *Finding:* Output any $v \in M_x$, with $M_x \neq \varnothing$.

Subroutines: $\forall x \in \mathcal{D}, v \in V$, $D_{v,x} \in \mathcal{S}$ *(data)*.

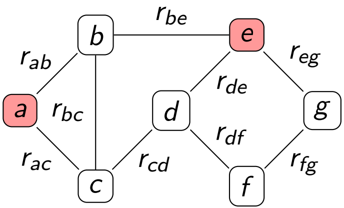
1. *Setup:* $v \mapsto D_{v,x}$ with S queries.
2. *Update:* $(vw, D_{v,x}) \mapsto D_{w,x}$, with U queries.
3. *Checking:* $(v, D_{v,x}) \mapsto [v \in M_x]$, with C queries.



$M_x = \{a, e\}$

Motivating example:
Element distinctness [Amb07]

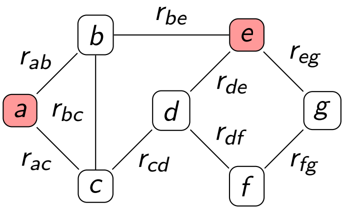
Random walk dynamics



$M_x = \{a, e\}$

Random walk dynamics

Canonical randomized algorithm:

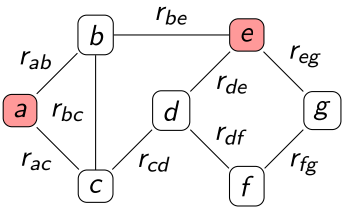


$M_x = \{a, e\}$

Random walk dynamics

Canonical randomized algorithm:

1. Sample $v \sim \sigma \in \Delta_V$.
2. Compute $D_{v,x}$. (cost S)

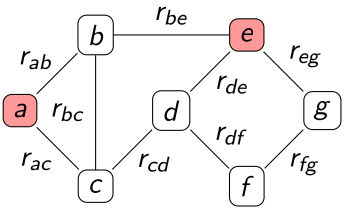


$M_x = \{a, e\}$

Random walk dynamics

Canonical randomized algorithm:

1. Sample $v \sim \sigma \in \Delta_V$.
2. Compute $D_{v,x}$. (cost S)
3. Repeat N_t times:
    1. If $v \in M_x$, output $\mathtt{yes}$. (cost C)

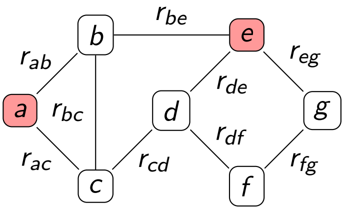


$M_x = \{a, e\}$

Random walk dynamics

Canonical randomized algorithm:

1. Sample $v \sim \sigma \in \Delta_V$.
2. Compute $D_{v,x}$. (cost S)
3. Repeat N_t times:
    1. If $v \in M_x$, output $\texttt{yes}$. (cost C)
    2. Repeat t times: *(fast-forwarding)*
        1. Sample $w \sim P_{v\cdot}$, where $P_{vw} \propto \frac{1}{r_{vw}}$.
        2. Compute $D_{w,x}$. (cost U)
        3. Set v to w.



$M_x = \{a, e\}$

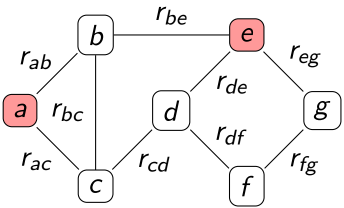
Random walk dynamics

Canonical randomized algorithm:

1. Sample $v \sim \sigma \in \Delta_V$.
2. Compute $D_{v,x}$. (cost S)
3. Repeat N_t times:
   1. If $v \in M_x$, output yes. (cost C)
   2. Repeat t times: *(fast-forwarding)*
      1. Sample $w \sim P_{v\cdot}$, where $P_{vw} \propto \frac{1}{r_{vw}}$.
      2. Compute $D_{w,x}$. (cost U)
      3. Set v to w.

Cost: $O(\mathsf{S} + N_t \cdot (t\mathsf{U} + \mathsf{C}))$.

1. *Detection:* $N_t \in \Theta(\max_{\substack{x \in \mathcal{D} \\ M_x \neq \varnothing}} \mathrm{HT}(P^t; \sigma \to M_x))$.
2. *Finding:* $N_t \in \Theta(\max_{x \in \mathcal{D}} \mathrm{HT}(P^t; \sigma \to M_x))$.



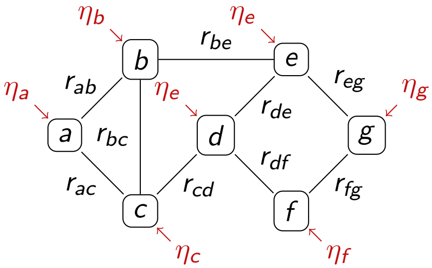
$M_x = \{a, e\}$

Effective resistance

Effective resistance

Definition:

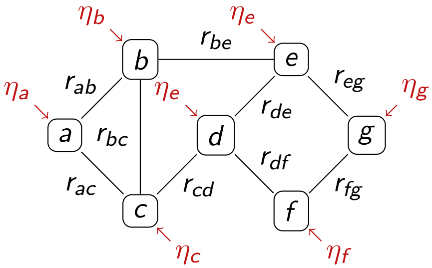
1. $\eta : V \to \mathbb{R}$ net-flow: $\sum_{v \in V} \eta_v = 0$.
   $R_{\text{eff}}(P; \eta) = \min_{\delta(f)=\eta} \sum_{e \in E} f_e^2 r_e$.



Effective resistance

Definition:

1. $\eta : V \to \mathbb{R}$ net-flow: $\sum_{v \in V} \eta_v = 0$.
   $R_{\text{eff}}(P; \eta) = \min_{\delta(f)=\eta} \sum_{e \in E} f_e^2 r_e$.
2. $\sigma \in \Delta_V$: $R_{\text{eff}}(P; \sigma \leftrightarrow M) = \min_{\nu \in \Delta_M} R_{\text{eff}}(P; \sigma - \nu)$.



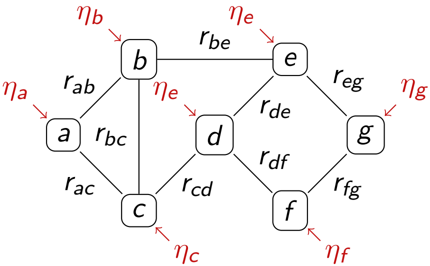
Effective resistance

Definition:

1. $\eta : V \to \mathbb{R}$ net-flow: $\sum_{v \in V} \eta_v = 0$.
   $R_{\mathrm{eff}}(P; \eta) = \min_{\delta(f)=\eta} \sum_{e \in E} f_e^2 r_e$.
2. $\sigma \in \Delta_V$: $R_{\mathrm{eff}}(P; \sigma \leftrightarrow M) = \min_{\nu \in \Delta_M} R_{\mathrm{eff}}(P; \sigma - \nu)$.

Relationships:

1. $\mathrm{HT}(P; \pi \to M_x) = R_{\mathrm{eff}}(P; \pi \leftrightarrow M_x)$.
2. $\mathrm{HT}(P; s \to t) + \mathrm{HT}(P; t \to s) = R_{\mathrm{eff}}(P; s \leftrightarrow t)$.



Wlog: $\sum_{e \in E} \frac{1}{r_e} = \frac{1}{2}$.

Effective resistance

Definition:

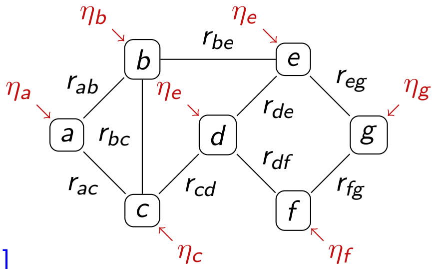
1. $\eta : V \to \mathbb{R}$ net-flow: $\sum_{v \in V} \eta_v = 0$.
   $R_{\text{eff}}(P; \eta) = \min_{\delta(f)=\eta} \sum_{e \in E} f_e^2 r_e$.
2. $\sigma \in \Delta_V$: $R_{\text{eff}}(P; \sigma \leftrightarrow M) = \min_{\nu \in \Delta_M} R_{\text{eff}}(P; \sigma - \nu)$.

Relationships:

1. $\mathrm{HT}(P; \pi \to M_x) = R_{\text{eff}}(P; \pi \leftrightarrow M_x)$.
2. $\mathrm{HT}(P; s \to t) + \mathrm{HT}(P; t \to s) = R_{\text{eff}}(P; s \leftrightarrow t)$.

Quantum algorithm: $\widetilde{O}(\mathsf{S} + \sqrt{1 + R_t} \cdot (\sqrt{t}\mathsf{U} + \mathsf{C}))$. [AGJ21]

1. *Detection:* $R_t \in \Theta(\max_{\substack{x \in \mathcal{D} \\ M_x \neq \varnothing}} R_{\text{eff}}(P^t; \sigma \leftrightarrow M_x))$.
2. *Finding:* $R_t \in \Theta(\max_{x \in \mathcal{D}} R_{\text{eff}}(P^t; \sigma \leftrightarrow M_x))$.



Wlog: $\sum_{e \in E} \frac{1}{r_e} = \frac{1}{2}$.

Quantum walk search – detection (similar to [Jef22])

Quantum walk search – detection (similar to [Jef22])

Subroutines:

1. S_v : compute $D_{v,x}$ with cost S.
   (function evaluation)
2. U_{vw} : update $D_{v,x} \mapsto D_{w,x}$ with cost U.
   (database update)
3. $C_{v,D}$: if $D = D_{v,x}$, compute $v \in M_x$ with cost C. *(on/off switch)*

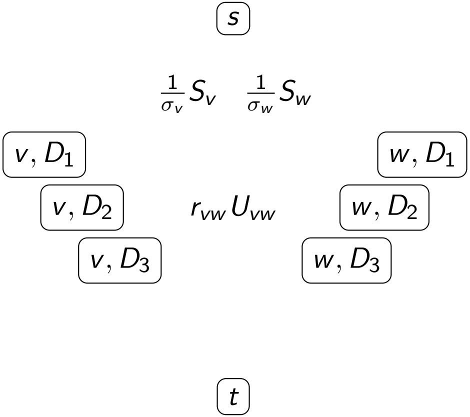
Quantum walk search – detection (similar to [Jef22])

Subroutines:

1. S_v : compute $D_{v,x}$ with cost S.
   (function evaluation)
2. U_{vw} : update $D_{v,x} \mapsto D_{w,x}$ with cost U.
   (database update)
3. $C_{v,D}$: if $D = D_{v,x}$, compute $v \in M_x$ with cost C. *(on/off switch)*

Hypergraph:

1. $V = \{(v, D) : v \in V, D \in \mathcal{S}\} \cup \{s, t\}$.



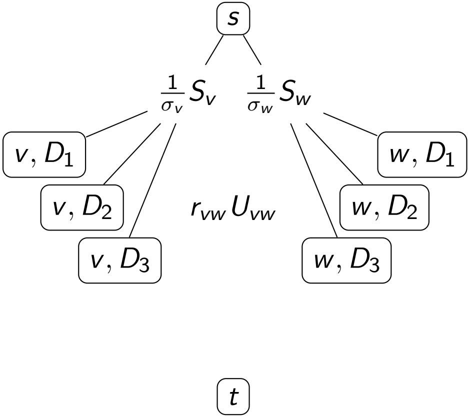
Quantum walk search – detection (similar to [Jef22])

Subroutines:

1. S_v : compute $D_{v,x}$ with cost S. *(function evaluation)*
2. U_{vw} : update $D_{v,x} \mapsto D_{w,x}$ with cost U. *(database update)*
3. $C_{v,D}$: if $D = D_{v,x}$, compute $v \in M_x$ with cost C. *(on/off switch)*

Hypergraph:

1. $V = \{(v, D) : v \in V, D \in \mathcal{S}\} \cup \{s, t\}$.



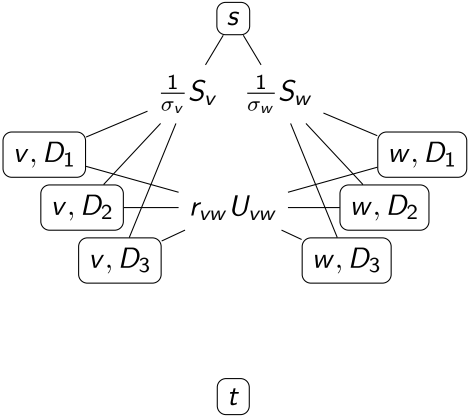
Quantum walk search – detection (similar to [Jef22])

Subroutines:

1. S_v : compute $D_{v,x}$ with cost S. *(function evaluation)*
2. U_{vw} : update $D_{v,x} \mapsto D_{w,x}$ with cost U. *(database update)*
3. $C_{v,D}$: if $D = D_{v,x}$, compute $v \in M_x$ with cost C. *(on/off switch)*

Hypergraph:

1. $V = \{(v, D) : v \in V, D \in \mathcal{S}\} \cup \{s, t\}$.



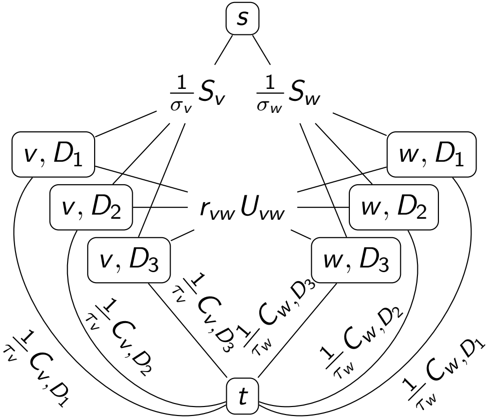
Quantum walk search – detection (similar to [Jef22])

Subroutines:

1. S_v : compute $D_{v,x}$ with cost S. *(function evaluation)*
2. U_{vw} : update $D_{v,x} \mapsto D_{w,x}$ with cost U. *(database update)*
3. $C_{v,D}$: if $D = D_{v,x}$, compute $v \in M_x$ with cost C. *(on/off switch)*

Hypergraph:

1. $V = \{(v, D) : v \in V, D \in \mathcal{S}\} \cup \{s, t\}$.



Quantum walk search – detection (similar to [Jef22])

Subroutines:

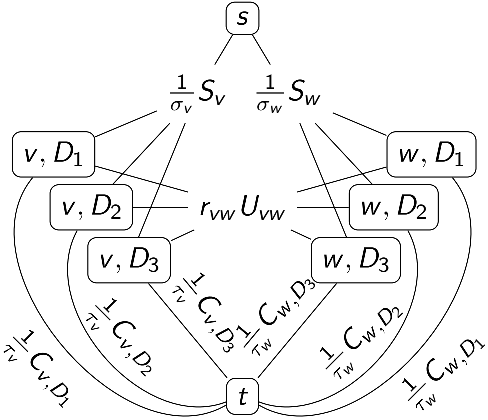
1. S_v : compute $D_{v,x}$ with cost S. *(function evaluation)*
2. U_{vw} : update $D_{v,x} \mapsto D_{w,x}$ with cost U. *(database update)*
3. $C_{v,D}$: if $D = D_{v,x}$, compute $v \in M_x$ with cost C. *(on/off switch)*

Hypergraph:

1. $V = \{(v, D) : v \in V, D \in \mathcal{S}\} \cup \{s, t\}$.

Parameters:

1. $\sigma, \tau \in \Delta_V$.
2. $\mu_x \in \Delta_V$, $\nu_x \in \Delta_{M_x}$.



Quantum walk search – detection (similar to [Jef22])

Subroutines:

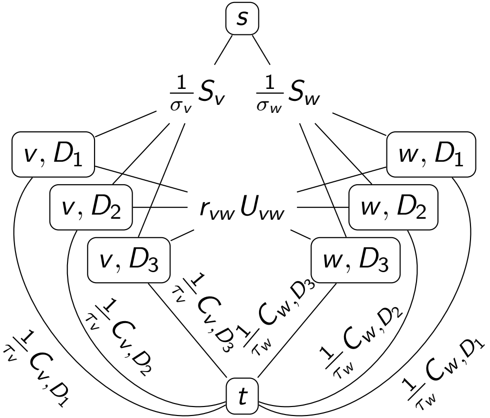
1. S_v : compute $D_{v,x}$ with cost S.
   (function evaluation)
2. U_{vw} : update $D_{v,x} \mapsto D_{w,x}$ with cost U.
   (database update)
3. $C_{v,D}$: if $D = D_{v,x}$, compute $v \in M_x$ with cost C. *(on/off switch)*

Hypergraph:

1. $V = \{(v, D) : v \in V, D \in \mathcal{S}\} \cup \{s, t\}$.

Parameters:

1. $\sigma, \tau \in \Delta_V$.
2. $\mu_x \in \Delta_V$, $\nu_x \in \Delta_{M_x}$.



$$\Rightarrow O\Big(\max_{\substack{x \in \mathcal{D} \\ M_x \neq \varnothing}} \sqrt{\sum_{v \in V} \frac{(\mu_x)_v^2}{\sigma_v}} \mathrm{S} + \sqrt{R_{\mathrm{eff}}(P; \mu_x - \nu_x)}\mathrm{U} + \sqrt{\sum_{v \in M_x} \frac{(\nu_x)_v^2}{\tau_v}} \mathrm{C}\Big) \text{ queries.}$$

Quantum walk search – detection analysis

$$O\Big(\max_{\substack{x \in \mathcal{D} \\ M_x \neq \varnothing}} \sqrt{\sum_{v \in V} \frac{(\mu_x)_v^2}{\sigma_v}\mathsf{S}} + \sqrt{R_{\mathrm{eff}}(P; \mu_x - \nu_x)}\mathsf{U} + \sqrt{\sum_{v \in M_x} \frac{(\nu_x)_v^2}{\tau_v}\mathsf{C}}\Big) \text{ queries.}$$

Quantum walk search – detection analysis

$$O\Big(\max_{\substack{x\in\mathcal{D}\\ M_x\neq\varnothing}} \sqrt{\sum_{v\in V} \frac{(\mu_x)_v^2}{\sigma_v}}\mathsf{S} + \sqrt{R_{\text{eff}}(P;\mu_x-\nu_x)}\mathsf{U} + \sqrt{\sum_{v\in M_x} \frac{(\nu_x)_v^2}{\tau_v}}\mathsf{C}\Big) \text{ queries.}$$

Plug in:

1. $\mu_x = \sigma$.
2. $\tau = (\sigma + \pi)/2$.

Quantum walk search – detection analysis

$$O\Big(\max_{\substack{x \in \mathcal{D} \\ M_x \neq \emptyset}} \sqrt{\sum_{v \in V} \frac{(\mu_x)_v^2}{\sigma_v}} \mathsf{S} + \sqrt{R_{\mathrm{eff}}(P; \mu_x - \nu_x)} \mathsf{U} + \sqrt{\sum_{v \in M_x} \frac{(\nu_x)_v^2}{\tau_v}} \mathsf{C}\Big) \text{ queries.}$$

Plug in:

1. $\mu_x = \sigma$.
2. $\tau = (\sigma + \pi)/2$.

Lemmas:

1. $R_{\mathrm{eff}}(P; \eta) \leq t \cdot R_{\mathrm{eff}}(P^t; \eta)$.
2. $\sum_{v \in V} \frac{(\nu_x)_v^2}{\tau_v} \leq 2(1 + R_{\mathrm{eff}}(P^t; \sigma - \nu))$.

Quantum walk search – detection analysis

$$O\Big(\max_{\substack{x\in\mathcal{D}\\M_x\neq\varnothing}}\sqrt{\sum_{v\in V}\frac{(\mu_x)_v^2}{\sigma_v}\mathsf{S}}+\sqrt{R_{\mathrm{eff}}(P;\mu_x-\nu_x)}\mathsf{U}+\sqrt{\sum_{v\in M_x}\frac{(\nu_x)_v^2}{\tau_v}\mathsf{C}}\Big)\text{ queries.}$$

Plug in:

1. $\mu_x = \sigma$.
2. $\tau = (\sigma + \pi)/2$.

Lemmas:

1. $R_{\mathrm{eff}}(P;\eta) \leq t \cdot R_{\mathrm{eff}}(P^t;\eta)$.
2. $\sum_{v\in V}\frac{(\nu_x)_v^2}{\tau_v} \leq 2(1 + R_{\mathrm{eff}}(P^t;\sigma-\nu))$.

Proof of first lemma:

1. $D = \mathrm{diag}(\pi) \quad \Rightarrow \quad L = D(I - P)$.
2. $\widetilde{\eta} = D^{-1/2}\eta$, $\widetilde{P} = D^{1/2}PD^{-1/2}$.
3. Observe that $\frac{1}{1-x} \leq t \cdot \frac{1}{1-x^t}$ $(x \in (-1,1))$.

$$\Rightarrow R_{\mathrm{eff}}(P;\eta) = \eta^T L^+\eta = \widetilde{\eta}^T(I-\widetilde{P})^+\widetilde{\eta}$$
$$\leq t\cdot\widetilde{\eta}^T(I-\widetilde{P}^t)\widetilde{\eta} = t\cdot R_{\mathrm{eff}}(P^t;\eta).\ \square$$

Quantum walk search – detection analysis

$$O\Big(\max_{\substack{x\in\mathcal{D}\\ M_x\neq\varnothing}} \sqrt{\sum_{v\in V}\frac{(\mu_x)_v^2}{\sigma_v}}\mathsf{S} + \sqrt{R_{\text{eff}}(P;\mu_x-\nu_x)}\mathsf{U} + \sqrt{\sum_{v\in M_x}\frac{(\nu_x)_v^2}{\tau_v}}\mathsf{C}\Big) \text{ queries.}$$

Plug in:

1. $\mu_x = \sigma$.
2. $\tau = (\sigma+\pi)/2$.

Lemmas:

1. $R_{\text{eff}}(P;\eta) \leq t \cdot R_{\text{eff}}(P^t;\eta)$.
2. $\sum_{v\in V}\frac{(\nu_x)_v^2}{\tau_v} \leq 2(1 + R_{\text{eff}}(P^t;\sigma-\nu))$.

Proof of first lemma:

1. $D = \mathrm{diag}(\pi) \quad \Rightarrow \quad L = D(I-P)$.
2. $\widetilde{\eta} = D^{-1/2}\eta$, $\widetilde{P} = D^{1/2}PD^{-1/2}$.
3. Observe that $\frac{1}{1-x} \leq t\cdot\frac{1}{1-x^t}$ $(x\in(-1,1))$.

$$\Rightarrow R_{\text{eff}}(P;\eta) = \eta^T L^+\eta = \widetilde{\eta}^T(I-\widetilde{P})^+\widetilde{\eta}$$
$$\leq t\cdot\widetilde{\eta}^T(I-\widetilde{P}^t)\widetilde{\eta} = t\cdot R_{\text{eff}}(P^t;\eta). \ \square$$

$$\Rightarrow O(\mathsf{S} + \sqrt{tR}\mathsf{U} + \sqrt{1+R}\mathsf{C}), \text{ where } R \in O\Big(\max_{\substack{x\in\mathcal{D}\\ M_x\neq\varnothing}} R_{\text{eff}}(P^t;\sigma\leftrightarrow M_x)\Big) \text{ queries.}$$

Quantum walk search – detection analysis

$$O\Big(\max_{\substack{x\in\mathcal{D}\\ M_x\neq\varnothing}} \sqrt{\sum_{v\in V}\frac{(\mu_x)_v^2}{\sigma_v}}\mathsf{S} + \sqrt{R_{\text{eff}}(P;\mu_x-\nu_x)}\mathsf{U} + \sqrt{\sum_{v\in M_x}\frac{(\nu_x)_v^2}{\tau_v}}\mathsf{C}\Big) \text{ queries.}$$

Plug in:

1. $\mu_x = \sigma$.
2. $\tau = (\sigma+\pi)/2$.

Lemmas:

1. $R_{\text{eff}}(P;\eta) \leq t\cdot R_{\text{eff}}(P^t;\eta)$.
2. $\sum_{v\in V}\frac{(\nu_x)_v^2}{\tau_v} \leq 2(1+R_{\text{eff}}(P^t;\sigma-\nu))$.

Proof of first lemma:

1. $D = \mathrm{diag}(\pi) \quad\Rightarrow\quad L = D(I-P)$.
2. $\widetilde{\eta} = D^{-1/2}\eta$, $\widetilde{P} = D^{1/2}PD^{-1/2}$.
3. Observe that $\frac{1}{1-x} \leq t\cdot\frac{1}{1-x^t}$ $(x\in(-1,1))$.

$$\Rightarrow R_{\text{eff}}(P;\eta) = \eta^T L^+\eta = \widetilde{\eta}^T(I-\widetilde{P})^+\widetilde{\eta}$$
$$\leq t\cdot\widetilde{\eta}^T(I-\widetilde{P}^t)\widetilde{\eta} = t\cdot R_{\text{eff}}(P^t;\eta). \;\square$$

$$\Rightarrow O(\mathsf{S}+\sqrt{tR}\mathsf{U}+\sqrt{1+R}\mathsf{C}), \text{ where } R\in O\Big(\max_{\substack{x\in\mathcal{D}\\ M_x\neq\varnothing}} R_{\text{eff}}(P^t;\sigma\leftrightarrow M_x)\Big) \text{ queries.}$$

Note: Does not use fast forwarding in the algorithm – merely in the analysis.

Summary & future research

Summary & future research

Summary:

1. State-reflection problems
2. Generalized graph composition
3. Quantum walk search construction
4. Improved analysis

Summary & future research

Summary:

1. State-reflection problems
2. Generalized graph composition
3. Quantum walk search construction
4. Improved analysis

Additional thoughts:

1. Amortization [CLM20; Jef22]

Summary & future research

Summary:

1. State-reflection problems
2. Generalized graph composition
3. Quantum walk search construction
4. Improved analysis

Additional thoughts:

1. Amortization [CLM20; Jef22]

Future directions:

1. Generalize quantum walk search to the finding setting.
2. Figure out quantum minimum finding.
3. Estimation problems?

Summary & future research

Summary:

1. State-reflection problems
2. Generalized graph composition
3. Quantum walk search construction
4. Improved analysis

Additional thoughts:

1. Amortization [CLM20; Jef22]

Future directions:

1. Generalize quantum walk search to the finding setting.
2. Figure out quantum minimum finding.
3. Estimation problems?

Thanks for you attention!

`ajcornelissen@outlook.com`

References

[AGJ21] Simon Apers, András Gilyén, and Stacey Jeffery. A unified framework of quantum walk search.

[Amb07] Andris Ambainis. Quantum walk algorithm for element distinctness.

[BJY24] Aleksandrs Belovs, Stacey Jeffery, Duyal Yolcu. Taming Quantum Time Complexity.

[CLM20] Titouan Carette, Mathieu Laurière, Frédéric Magniez. Extended Learning Graphs for Triangle Finding.

[Jef22] Stacey Jeffery. Quantum Subroutine Composition.

[LMR+11] Troy Lee, Rajat Mittal, Ben W Reichardt, Robert Špalek, and Mario Szegedy. Quantum query complexity of state conversion.