Quantum Algorithms for Graph Connectivity and Formula Evaluation

Stacey Jeffery[1] and Shelby Kimmel[2]

[1]QuSoft and CWI, Amsterdam, the Netherlands
[2]Middlebury College, Middlebury, VT, USA

August 17, 2017

$O(\sqrt{N})$ N $O(|V|^{3/2})$ V

1 Introduction

Stacey Jeffery: jeffery@cwi.nl, SJ completed parts of this work while at the Institute for Quantum Information and Matter (IQIM), Caltech

Shelby Kimmel: shelby.kimmel@gmail.com, SK completed parts of this work while at the Joint Center for Quantum Information and Computer Science (QuICS), University of Maryland

st-

NAND $O(\sqrt{N})$ N NAND $\Theta(N^{.753})$ $\Omega(N^{.51})$ NAND *k-fault trees*, $O(2^k)$, $\Omega((\log \frac{\log N}{k})^k)$, *superpolynomial* k.

Contributions.

- G $G \cup \{\{s,t\}\}$ —
  - –
  - –
- N $O(\sqrt{N})$
  - –
  - –

Open Problems.

Organization:

2 Preliminaries

2.1 Graph Theory

Definition 1 (Unit *st*-flow)**.** *Let G be an undirected weighted graph with $s, t \in V(G)$, and s and t connected. Then a* unit *st*-flow *on G is a function $\theta : \overrightarrow{E}(G) \to \mathbb{R}$ such that:*

1. *For all $(u, v, \lambda) \in \overrightarrow{E}(G)$, $\theta(u, v, \lambda) = -\theta(v, u, \lambda)$;*
2. *$\sum_{v,\lambda:(s,v,\lambda)\in\overrightarrow{E}} \theta(s, v, \lambda) = \sum_{v,\lambda:(v,t,\lambda)\in\overrightarrow{E}} \theta(v, t, \lambda) = 1$; and*
3. *for all $u \in V(G) \setminus \{s, t\}$, $\sum_{v,\lambda:(u,v,\lambda)\in\overrightarrow{E}} \theta(u, v, \lambda) = 0$.*

Definition 2 (Unit Flow Energy)**.** *Given a unit st-flow θ on a graph G, the* unit flow energy *is*

$$J(\theta) = \sum_{(\{u,v\},\lambda)\in E(G)} \theta(u,v,\lambda)^2. \tag{1}$$

Definition 3 (Effective resistance)**.** *Let G be a graph with $s,t \in V(G)$. If s and t are connected in G, the* effective resistance *is $R_{s,t}(G) = \min_\theta J(\theta)$, where θ runs over all unit st-flows. If s and t are not connected, $R_{s,t}(G) = \infty$.*

$R_{s,t}(G)$ s t s t G, $R_{s,t}(G)|E(G)|$ *commute time* s t s t s G e G s t, $R_{s,t}(G)$

network G *weight* $c : E(G) \to \mathbb{R}^+$.

Definition 4 (Effective Resistance with weights)**.** *Let $\mathcal{N} = (G,c)$ be a network with $s,t \in V(G)$. The* effective resistance of $\mathcal{N}$ *is $R_{s,t}(\mathcal{N}) = \min_\theta \sum_{(\{u,v\},\lambda)\in E(G)} \frac{\theta(u,v,\lambda)^2}{c(\{u,v\},\lambda)}$, where θ runs over all unit st-flows.*

u $(\{u,v\},\lambda)$ $c(\{u,v\},\lambda)$. s t $R_{s,t}(\mathcal{N}) \sum_{e\in E(G)} c(e)$. $\mathcal{N}$ e $1/c(e)$, $R_{s,t}(\mathcal{N})$ G $e = (\{s,t\},\lambda)$ $c(e)$, $R_{s,t}(G) = 1/c(e)$. $R_{s,t}$,

conductance e $c(e)$,

Claim 5. *Let two networks $\mathcal{N}_1 = (G_1,c_1)$ and $\mathcal{N}_2 = (G_2,c_2)$ each have nodes s and t. If we create a new graph G by identifying the s nodes and the t nodes (i.e. connecting the graphs in parallel) and define $c : E(G) \to \mathbb{R}^+$ by $c(e) = c_1(e)$ if $e \in E(G_1)$ and $c(e) = c_1(e)$ if $e \in E(G_2)$, then*

$$R_{s,t}(G,c) = \left(\frac{1}{R_{s,t}(G_1,c_1)} + \frac{1}{R_{s,t}(G_2,c_2)}\right)^{-1}. \tag{2}$$

However, if we create a new graph G by identifying the t node of G_1 with the s node of G_2, relabeling this node $v \notin \{s,t\}$ (i.e. connecting the graphs in series) and define c as before, then

$$R_{s,t}(G,c) = R_{s,t}(G_1,c_1) + R_{s,t}(G_2,c_2). \tag{3}$$

$R_{s,t}(G_1, c_1)$ $R_{s,t}(G_2, c_2)$ FALSE, TRUE, (3) OR, $0+0=0$, $0+\infty=\infty+0=\infty+\infty=\infty$. (2) AND $\frac{1}{0}=\infty$ $\frac{1}{\infty}=0$.

Definition 6 (st-cut)**.** *Given a graph G with $s,t \in V(G)$, if s and t are not connected, an st-*cut *is a function $\kappa : V(G) \to \{0,1\}$ such that $\kappa(s)=1$, $\kappa(t)=0$, and $\kappa(v)-\kappa(u)=0$ whenever $\{u,v\} \in E(G)$.*

κ $S \subset V(G)$ $s \in S$, $t \notin S$, G S, $\overline{S}$. s t G, s t.

Definition 7 (Dual Graph)**.** *Let G be a planar graph (with an implicit embedding). The* dual *graph, $G^\dagger$, is defined as follows. For every face $f \in F(G)$, $G^\dagger$ has a vertex v_f, and any two vertices are adjacent if their corresponding faces share an edge, e. We call the edge between two such vertices the* dual edge *to e, $e^\dagger$. By convention, e and $e^\dagger$ will always have the same label, so that if $e=(\{u,v\},\lambda)$, then $e^\dagger=(\{v_f,v_{f'}\},\lambda)$ for f and f' the faces of G on either side of the edge e.*

2.2 Span Programs and Quantum Query Algorithms

Definition 8 (Span Program)**.** *A span program $P=(H,U,\tau,A)$ on $\{0,1\}^N$ is made up of* **(I)** *finite-dimensional inner product spaces $H=H_1\oplus\cdots\oplus H_N$, and $\{H_{j,b}\subseteq H_j\}_{j\in[N],b\in\{0,1\}}$ such that $H_{j,0}+H_{j,1}=H_j$,* **(II)** *a vector space U,* **(III)** *a non-zero* target vector *$\tau \in U$, and* **(IV)** *a linear operator $A : H \to U$. For every string $x \in \{0,1\}^N$, we associate the subspace $H(x) := H_{1,x_1}\oplus\cdots\oplus H_{N,x_N}$, and an operator $A(x) := A\Pi_{H(x)}$, where $\Pi_{H(x)}$ is the orthogonal projector onto $H(x)$.*

Definition 9 (Positive and Negative Witness)**.** *Let P be a span program on $\{0,1\}^N$ and let x be a string $x \in \{0,1\}^N$. Then we call $|w\rangle$ a* positive witness for x in P *if $|w\rangle \in H(x)$, and $A|w\rangle=\tau$. We define the* positive witness size of x *as:*

$$w_+(x,P)=w_+(x)=\min\{\||w\rangle\|^2 : |w\rangle \in H(x), A|w\rangle=\tau\}, \tag{4}$$

if there exists a positive witness for x, and $w_+(x)=\infty$ otherwise. Let $\mathcal{L}(U,\mathbb{R})$ denote the set of linear maps from U to $\mathbb{R}$. We call a linear map $\omega \in \mathcal{L}(U,\mathbb{R})$ a negative witness for x in P *if $\omega A\Pi_{H(x)}=0$ and $\omega\tau=1$. We define the* negative witness size of x *as:*

$$w_-(x,P)=w_-(x)=\min\{\|\omega A\|^2 : \omega \in \mathcal{L}(U,\mathbb{R}), \omega A\Pi_{H(x)}=0, \omega\tau=1\}, \tag{5}$$

if there exists a negative witness, and $w_-(x)=\infty$ otherwise. If $w_+(x)$ is finite, we say that x is positive *(wrt. P), and if $w_-(x)$ is finite, we say that x is* negative*. We let P_1 denote the set of positive inputs, and P_0 the set of negative inputs for P. In this way, the span program defines a partition (P_0,P_1) of $[N]$.*

$f : X \to \{0,1\}$, $X \subseteq \{0,1\}^N$, *P decides f* $f^{-1}(0) \subseteq P_0$ $f^{-1}(1) \subseteq P_1$. P f, $x \in X$ $\mathcal{O}_x : |i,b\rangle \mapsto |i,b\oplus x_i\rangle$.

Theorem 10 ([21]). *Fix $X \subseteq \{0,1\}^N$ and $f : X \to \{0,1\}$, and let P be a span program on $\{0,1\}^N$ that decides f. Let $W_+(f,P) = \max_{x \in f^{-1}(1)} w_+(x,P)$ and $W_-(f,P) = \max_{x \in f^{-1}(0)} w_-(x,P)$. Then there is a bounded error quantum algorithm that decides f with quantum query complexity $O(\sqrt{W_+(f,P)W_-(f,P)})$.*

2.3 Boolean Formulas

A
$x_1, \ldots, x_N$,
$\{\wedge, \vee, \neg\}$.
$\neg$ (NOT),
$\wedge$ (AND)
$\vee$ (OR),
fan-in
depth
AND-OR *formula*
monotone formula
$\wedge$
$\vee$.
AND-OR
NOT
NOT
formula,
AND-OR *read-once formula*.
x_i
$x \in \{0,1\}^N$
ϕ,
$\phi(x)$
ϕ
$\phi = x_i$
$\phi(x) = x_i$.
ϕ
ϕ
$\phi_1, \ldots, \phi_l$,
$\phi = \phi_1 \wedge \cdots \wedge \phi_l$,
$\wedge$,
$\phi = \phi_1 \vee \cdots \vee \phi_l$,
$\vee$.
$\phi(x) = \phi_1(x) \wedge \cdots \wedge \phi_l(x)$,
$\phi(x) = \phi_1(x) \vee \cdots \vee \phi_l(x)$.
$\phi = \phi_N$
N
$\textsc{Eval}_\phi$,
$x \in \{0,1\}^N$,
$\phi_N(x)$.
$\phi(x) = 0$,
x is a 0-
$\phi(x) = 1$, x is a
$\phi_1 \circ \phi_2$,
ϕ_1
ϕ_2.
$\phi_1 : \{0,1\}^{N_1} \to \{0,1\}$
$\phi_2 : \{0,1\}^{N_2} \to \{0,1\}$,
$\phi_1 \circ \phi_2 : \{0,1\}^{N_1 N_2} \to \{0,1\}$
$\phi_1 \circ \phi_2(x) = \phi_1(\phi_2(x^1), \ldots, \phi_2(x^{N_1}))$,
$x = (x^1, \ldots, x^{N_1})$
$x^i \in \{0,1\}^{N_2}$.
NAND
NAND
d —
d
$\vee$
$\wedge$
NAND_d
NAND
d.
NAND_d
d,
d
AND
OR
NAND_d
$x \in \{0,1\}^N$,
$N = 2^d$.
$\text{NAND}_2(x_1, x_2, x_3, x_4) = (x_1 \wedge x_2) \vee (x_3 \wedge x_4)$
NAND
2. NAND_0
A NAND_d
$x \in \{0,1\}^{2^d}$
NAND_d,
x_i,
A
B)
1,
A
0,
B

Figure 1: Example of how to derive $\overline{G}$, $\overline{G}^\dagger$, and G' from a planar graph G where s and t are on the same face. $\overline{G}$ is obtained from G by adding an edge $(\{s,t\},\emptyset)$. $\overline{G}^\dagger$ is the planar dual of $\overline{G}$. (In the diagram labeled by $\overline{G}^\dagger$, $\overline{G}$ is the gray graph, while $\overline{G}^\dagger$ is black). G' is obtained from $\overline{G}^\dagger$ by removing the edge $(\{s,t\},\emptyset)^\dagger$. Note that dual edges inherit their labels (in this case $1,2,3,4,\emptyset$) from the primal edge.

Lemma 11. *Let G be a planar multigraph with $s,t \in V(G)$ such that $G \cup \{\{s,t\}\}$ is also planar, and let c be a weight function on $E(G)$. Let $x \in \{0,1\}^{E(G)}$. Then $w_+(x, P_{G,c}) = \frac{1}{2}R_{s,t}(G(x),c)$ and $w_-(x, P_{G,c}) = 2R_{s',t'}(G'(x),c')$.*

Theorem 12. *Let G be a planar multigraph with $s,t \in V(G)$ such that $G \cup \{\{s,t\}\}$ is also planar. Then the bounded error quantum query complexity of evaluating st-$\mathrm{CONN}_{G,D}$ is*

$$O\left(\min_c \sqrt{\max_{x\in D: st\text{-}\mathrm{CONN}_G(x)=1} R_{s,t}(G(x),c) \times \max_{x\in D: st\text{-}\mathrm{CONN}_G(x)=0} R_{s',t'}(G'(x),c')}\right) \tag{10}$$

where the minimization is over all positive real-valued functions c on $E(G)$.

$$U_{G,c} : |u\rangle|0\rangle \mapsto \frac{1}{\sqrt{\sum_{v,\lambda:(u,v,\lambda)\in\vec{E}(G)} c(\{u,v\},\lambda)}} \sum_{v,\lambda:(u,v,\lambda)\in\vec{E}(G)} \sqrt{c(\{u,v\},\lambda)}|u\rangle|u,v,\lambda\rangle. \tag{11}$$

Theorem 13. *Let $P_{G,c} = (H, U, A, \tau)$ be defined as in (6). Let $S_{G,c}$ be an upper bound on the time complexity of implementing $U_{G,c}$. If G has the property that $G \cup \{\{s,t\}\}$ is planar, then the time complexity of deciding st-CONN$_{G,D}$ is at most*

$$O\left(\min_c S_{G,c}\sqrt{\max_{x\in D: s,t \text{ are connected}} R_{s,t}(G(x),c) \times \max_{x\in D: s,t \text{ are not connected}} R_{s',t'}(G'(x),c')}\right). \tag{12}$$

$U_{G,c}$ in $S_{G,c}$ is $S'_{G,c}$, $O(\max\{\log|E(G)|, \log|V(G)|\} + S'_{G,c})$.

3.1 Comparison to Previous Quantum Algorithm

$G \cup \{\{s,t\}\}$ (10) (8), c 1

$$C_{s,t}(G(x)) = (\text{shortest path length from } s' \text{ to } t' \text{ in } G'(x)) \geq R_{s't'}(G'(x)). \tag{13}$$

(13), $s't'$ $G'(x)$ s' t' $C_{s,t}(G(x))$ $R_{s',t'}(G(x))$

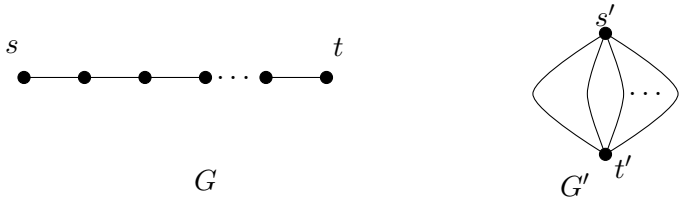


Figure 2: Example of graph for which our analysis does better than the analysis of [4], even with $c = 1$ for all edges, under the promise that $G'(x)$ always contains at least h edges, if s' and t' are connected.

$$\max_{x \in D : st\text{-}\mathrm{CONN}_G(x)=1} R_{s,t}(G(x)) = N \tag{14}$$

$$\max_{x \in D : st\text{-}\mathrm{CONN}_G(x)=0} R_{s',t'}(G'(x)) \leq 1/h, \tag{15}$$

$$\max_{x \in D : st\text{-}\mathrm{CONN}_G(x)=0} C_{s,t}(G(x)) = 1. \tag{16}$$

$$\max_{x \in D : st\text{-}\mathrm{CONN}_G(x)=1} R_{s,t}(G(x), c) = 2N, \tag{17}$$

$$\max_{x \in D : st\text{-}\mathrm{CONN}_G(x)=0} R_{s',t'}(G'(x), c') \leq 1, \tag{18}$$

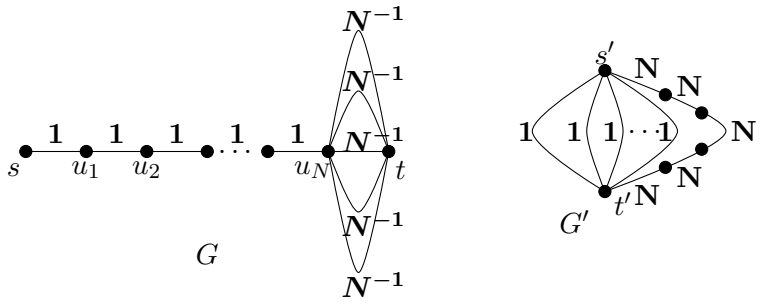


Figure 3: Example of graph for which our analysis does quadratically better than the analysis of [4] by taking advantage of a non-trivial weight function c. The values of c for each edge of G, and of c' for each edge of G', are shown in boldface.

[illegible] s' to t'. [illegible]
N [illegible] N, [illegible] $1/N$ [illegible]

$$\max_{x \in D : st\text{-}\mathrm{CONN}_G(x)=0} C_{s,t}(G(x)) = N \tag{19}$$

[illegible] u_N and t, and

$$\max_{x \in D : st\text{-}\mathrm{CONN}_G(x)=1} R_{s,t}(G(x)) = N + 1, \tag{20}$$

[illegible] u_N and t is [illegible] $O(N)$ [illegible]
[illegible] $O(N^{1/2})$.
[illegible]

4 AND-OR Formulas and st-Connectivity

[illegible] AND-OR [illegible] *st*-[illegible] [illegible] $x = (x_1, \ldots, x_N)$ [illegible]
[illegible] O_x, [illegible] $O_x|i\rangle|b\rangle = |i\rangle|b \oplus x_i\rangle$.
[illegible] AND-OR [illegible] ϕ [illegible] N [illegible]
[illegible] G_ϕ, [illegible] G_ϕ [illegible] s and t [illegible]
[illegible] G_ϕ [illegible] $\{x_i\}_{i\in[N]}$. [illegible] $\phi = x_i$ [illegible]
[illegible] G_ϕ [illegible] s and t, [illegible] x_i.
[illegible] $E(G_\phi) = \{(\{s,t\}, x_i)\}$ and $V(G_\phi) = \{s,t\}$.
[illegible] $\phi = \phi_1 \wedge \cdots \wedge \phi_l$. [illegible] G_ϕ [illegible]
$G_{\phi_1}, \ldots, G_{\phi_l}$ [illegible] t in G_{ϕ_i} [illegible] s in $G_{\phi_{i+1}}$,
[illegible] $i = 1, \ldots, l-1$, [illegible] s in G_{ϕ_1} by s, [illegible]
t in G_{ϕ_l} by t. [illegible] $G_{\phi_1}, \ldots, G_{\phi_l}$ *in series*, [illegible]
[illegible] G_ϕ, [illegible]
[illegible] $\phi = \phi_1 \vee \cdots \vee \phi_l$. [illegible] G_ϕ
[illegible] $G_{\phi_1}, \ldots, G_{\phi_l}$ [illegible] s, [illegible]

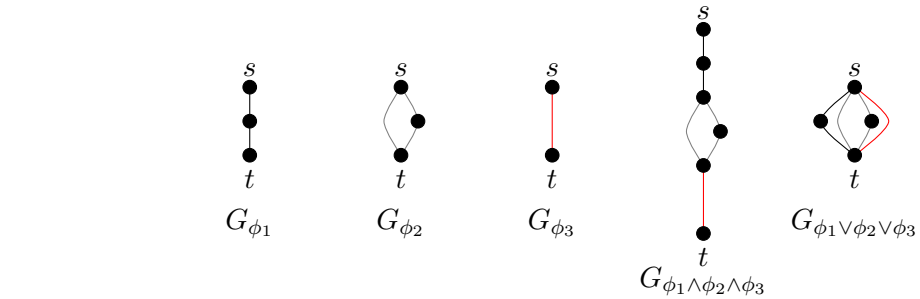


Figure 4: Let $\phi_1 = x_1 \wedge x_2$, $\phi_2 = x_3 \vee (x_4 \wedge x_5)$, and $\phi_3 = x_6$. Then we obtain $G_{\phi_1 \wedge \phi_2 \wedge \phi_3}$ by connecting G_{ϕ_1}, G_{ϕ_2}, and G_{ϕ_3} in series, and $G_{\phi_1 \vee \phi_2 \vee \phi_3}$ by connecting them in parallel.

Lemma 14. *Let ϕ be any* AND-OR *formula on N variables. For every $x \in \{0,1\}^N$, there exists a path from s to t in $G_\phi(x)$ if and only if $\phi(x) = 1$. Furthermore, for every $x \in \{0,1\}^N$, there exists a path from s' to t' in $G'_\phi(x)$ if and only if $\phi(x) = 0$.*

Theorem 15. *For any family ϕ of* AND-OR *formulas, the bounded error quantum query complexity of* $\textsc{Eval}_\phi$ *when the input is promised to come from a set D is*

$$O\left(\min_c \sqrt{\max_{x \in D:\phi(x)=1} R_{s,t}(G_\phi(x), c) \times \max_{x \in D:\phi(x)=0} R_{s',t'}(G'_\phi(x), c')}\right), \tag{21}$$

where the minimization is over all positive real-valued functions c on $E(G_\phi)$.

Proof. By Lemma 14, the query complexity of $\textsc{Eval}_\phi$ on D is at most the query complexity of st-$\textsc{conn}_{G_\phi,D}$. Since G_ϕ is planar, and has s and t on the same face, we can apply Theorem 12, which immediately implies the result. □

4.1 Comparison to Existing Boolean Formula Algorithms

N

$O(\sqrt{N})$

Theorem 16. *Let ϕ be a read-once formula on N variables. Then there exists a choice of c on $E(G_\phi)$ such that the quantum algorithm obtained from the span program $P_{G_\phi,c}$ computes* $\textsc{Eval}_\phi$ *with bounded error in $O(\sqrt{N})$ queries.*

Claim 17. *If $\phi = \phi_1 \vee \phi_2 \vee \cdots \vee \phi_l$, then $G'_\phi(x)$ is formed by composing $\{G'_{\phi_i}(x)\}_i$ in series, and if $\phi = \phi_1 \wedge \phi_2 \wedge \cdots \wedge \phi_l$, then $G'_\phi(x)$ is formed by composing $\{G'_{\phi_i}(x)\}_i$ in parallel.*

G'_ϕ

G'_ϕ

ϕ

N

ϕ'

N

$x \in \{0,1\}^N$, $\phi(x) = \neg\phi'(\bar{x})$,

$\bar{x}$

x.

$G_{\phi'} = G'_\phi$,

x, $G_{\phi'}(\bar{x}) = G'_\phi(x)$

Proof of Theorem 16. We will make use of the following fact: for any network (G, c), and any positive real number W:

$$R_{s,t}(G, c/W) = \min_\theta \sum_{e\in E(G)} \frac{\theta(e)^2}{c(e)/W} = W \min_\theta \sum_{e\in E(G)} \frac{\theta(e)^2}{c(e)} = W R_{s,t}(G, c). \tag{22}$$

We now proceed with the proof. For any formula ϕ in $\{\wedge, \vee, \neg\}$, by repeated applications of de Morgan's law, we can push all NOT-gates to distance-1 from a leaf. Since x_i and $\neg x_i$ can both be learned in one query, we can restrict our attention to AND-OR formulas.

If ϕ has only $N = 1$ variable, it's easy to see that $W_+(P_{G_\phi,c})W_-(P_{G_\phi,c}) \leq N$ for c taking value 1 on the single edge in G_ϕ. We will prove by induction that this is true for any ϕ, for some choice of c, completing the proof, since the complexity of our algorithm obtained from $P_{G_\phi,c}$ is $O\left(\sqrt{W_+(P_{G_\phi,c})W_-(P_{G_\phi,c})}\right)$.

Suppose $\phi = \phi_1 \wedge \cdots \wedge \phi_l$ for formulas ϕ_i on N_i variables, so ϕ has $N = \sum_i N_i$ variables. For $x \in \{0,1\}^N$, we will let $x^i \in \{0,1\}^{N_i}$ denote the $(N_1 + \cdots + N_{i-1} + 1)$-th to $(N_1 + \cdots + N_i)$-th bits of x. For each G_{ϕ_i}, by the induction hypothesis, there is some weight function c_i on $E(G_{\phi_i})$ such that $W_+(P_{G_{\phi_i},c_i})W_-(P_{G_{\phi_i},c_i}) \leq N_i$.

Using our construction, G_ϕ is formed by composing $\{G_{\phi_i}\}_i$ in series. Thus every edge $(\{u, v\}, \lambda) \in E(G_\phi)$ corresponds to an edge $(\{u, v\}, \lambda) \in E(G_{\phi_i})$ for some i. We create a weight function $c : E(G_\phi) \to \mathbb{R}^+$ such that $c(\{u, v\}, \lambda) = \frac{c_i(\{u,v\},\lambda)}{W_-(P_{G_{\phi_i},c_i})}$ if $(\{u, v\}, \lambda)$ is an edge originating from the graph G_{ϕ_i}. That is, our new weight function is the same as combining all of the old weight functions, up to scaling factors $\{W_-(P_{G_{\phi_i},c_i})\}_i$.

Using Lemma 11, Claim 5, and Eq. (22), for any 1-instance x,

$$\begin{aligned} w_+(x, P_{G_phi,c}) &= \frac{1}{2} R_{s,t}(G_\phi(x), c) = \frac{1}{2} \sum_{i=1}^{l} R_{s,t}\left(G_{\phi_i}(x), \frac{c_i}{W_-(P_{G_{\phi_i},c_i})}\right) \\ &= \frac{1}{2} \sum_{i=1}^{l} W_-(P_{G_{\phi_i},c_i}) R_{s,t}\left(G_{\phi_i}(x), c_i\right) \leq \sum_{i=1}^{l} W_-(\phi_i, P_{G_{\phi_i},c_i}) W_+(P_{G_{\phi_i},c_i}). \end{aligned} \tag{23}$$

Thus

$$W_+(P_{G_\phi,c}) \leq \sum_{i=1}^{l} W_-(P_{G_{\phi_i},c_i}) W_+(P_{G_{\phi_i},c_i}) \leq \sum_{i=1}^{l} N_i = N. \tag{24}$$

Recall that for a weight function c on G_ϕ, we define a weight function c' on G'_ϕ by $c'(e^\dagger) = 1/c(e)$. Then for an edge $e \in E(G_{\phi_i}(x^i))$, we have $c'(e^\dagger) = W_-(P_{G_{\phi_i},c_i})/c_i(e) = W_-(P_{G_{\phi_i},c_i}) c'_i(e^\dagger)$. By Claim 17, G'_ϕ is formed by composing $\{G'_{\phi_i}\}_i$ in parallel, so by Lemma 11, Claim 5, and Eq. (22):

$$\begin{aligned} w_-(x, P_{G_\phi,c}) &= 2R_{s',t'}(G'_\phi(x), c') = 2\left(\sum_{i=1}^{l} \frac{1}{R_{s',t'}(G'_{\phi_i}(x^i), c')}\right)^{-1} \\ &= 2\left(\sum_{i=1}^{l} \frac{1}{R_{s',t'}(G'_{\phi_i}(x^i), W_-(P_{G_{\phi_i},c_i}) c'_i)}\right)^{-1} = 2\left(\sum_{i=1}^{l} \frac{W_-(P_{G_{\phi_i},c_i})}{R_{s',t'}(G'_{\phi_i}(x^i), c'_i)}\right)^{-1} \\ &= 2\left(\sum_{i=1}^{l} \frac{W_-(P_{G_{\phi_i},c_i})}{w_-(x^i, P_{G_{\phi_i},c_i})}\right)^{-1}. \end{aligned} \tag{25}$$

Whenever x is a 0-instance of ϕ, the set $S \subseteq [l]$ of i such that x^i is a 0-instance of ϕ_i is non-empty. This is exactly the set of i such that $w_-(x^i, P_{G_{\phi_i},c_i}) < \infty$. Continuing from above, we have:

$$w_-(x, P_{G_\phi,c}) = \left(\sum_{i\in S} \frac{W_-(P_{G_{\phi_i},c_i})}{w_-(x^i, P_{G_{\phi_i},c_i})}\right)^{-1} \leq \left(\sum_{i\in S} \frac{W_-(P_{G_{\phi_i},c_i})}{W_-(P_{G_{\phi_i},c_i})}\right)^{-1} = \frac{1}{|S|} \leq 1. \tag{26}$$

Thus $W_-(P_{G_{\phi_i},c_i}) \leq 1$. Combining this with Eq. (24) we have $W_+(P_{G_\phi,c}) W_-(P_{G_\phi,c}) \leq N$, as desired.

The proof for the case $\phi = \phi_1 \vee \cdots \vee \phi_l$ is similar. □

Corollary 18. *Deciding st-connectivity on subgraphs of two-terminal series-parallel graphs of N edges can be accomplished using $O(\sqrt{N})$ queries, if s and t are chosen to be the two terminal nodes.*

$N, h \in \mathbb{Z}^+$, $D_{N,h} = \{x \in \{0,1\}^N : |x| = N \text{ or } |x| \leq N - h\}$ $D'_{N,h} = \{x \in \{0,1\}^N : |x| = 0 \text{ or } |x| \geq h\}$.

AND-OR AND) OR) AND $|_{D_{N,h}}$ $D'_{N,h}$ OR $|_{D'_{N,h}}$ OR 1, h.

Theorem 19. *Let $\phi = \phi_1 \circ \phi_2 \circ \cdots \circ \phi_l$, where for each $i \in [l]$, $\phi_i = \text{OR}|_{D'_{N_i,h_i}}$ or $\phi_i = \text{AND}|_{D_{N_i,h_i}}$. Then the randomized bounded-error query complexity of evaluating ϕ is $\Omega\left(\prod_{i=1}^{l} N_i/h_i\right)$, and the bounded-error quantum query complexity of evaluating ϕ is $O\left(\prod_{i=1}^{l} \sqrt{N_i/h_i}\right)$.*

[illegible] ϕ_1 [illegible] $D_1 \subseteq \{0,1\}^{N_1}$ [illegible] ϕ_2 [illegible] $D_2 \subseteq \{0,1\}^{N_2}$, $\phi = \phi_1 \circ \phi_2$ [illegible] $x = (x^1, \ldots, x^{N_1}) \in \{0,1\}^{N_1 N_2}$ [illegible] $x^i \in D_2$ [illegible] $i \in [N_1]$, [illegible] $(\phi_2(x^1), \ldots, \phi_2(x^{N_1})) \in D_1$.

[illegible]

$$\frac{\prod_{i=1}^{l} N_i}{\prod_{i=1}^{l} h_i} = \left(\max_{x \in D:\phi(x)=1} R_{s,t}(G_\phi(x))\right)\left(\max_{x \in D:\phi(x)=0} R_{s,t}(G'_\phi(x))\right), \tag{27}$$

[illegible] ϕ. [illegible]

[illegible] *st*- [illegible] $\text{AND}|_{D_{N,h}}$, [illegible] *st*-

5 NAND-tree Results

5.1 Query Separations

[illegible] NAND [illegible] k [illegible] F_k^d [illegible] NAND_d [illegible] k [illegible]

Theorem 20. *Using the st-connectivity approach to formula evaluation (Theorem 15), one can solve $\text{EVAL}_{\text{NAND}_d}$ when the input is promised to be from $F^d_{\log d}$ with $O(d)$ queries, while any classical algorithm requires $\Omega(d^{\log\log(d)})$ queries.*

[illegible] k, [illegible] *st*- [illegible]

Theorem 21. *Consider the problem st-$\text{CONN}_{G_{\text{NAND}_d},F_1^d}$. The analysis of [4] gives a bound of $O(N^{1/4})$ quantum queries to decide this problem (where $N = 2^d$ is the number of edges in G_{NAND_d}), while our analysis shows this problem can be decided with $O(1)$ quantum queries.*

[illegible] k [illegible] NAND [illegible] NAND [illegible]

NAND_d. $G_{\mathrm{NAND}_d}(x)$ $G'_{\mathrm{NAND}_d}(x)$.

NAND_d $x \in \{0,1\}^{2^d}$, NAND-

NAND A B p

NAND x is Z

$\mathcal{P}_Z(x)$ Z Z

$\mathcal{P}_A(x)$ $\mathcal{P}_B(x)$

A B

B A

A B

$\mathcal{P}_A$. $\nu_Z(p)$

p Z $\nu_A(p)$ $\nu_B(p))$

p > 0

v a *fault* A

B

$f_Z(p)$ $\nu_Z(p)$,

$\mathcal{F}(x)$ x [1] $\mathcal{F}(x) = \min\{\mathcal{F}_A(x), \mathcal{F}_B(x)\}$,

$$\mathcal{F}_Z(x) = \begin{cases} 2^{\max_{p \in \mathcal{P}_Z(x)} f_Z(p)} & \text{if } x \text{ is } Z\text{-winnable} \\ \infty & \text{otherwise.} \end{cases} \tag{28}$$

$k = 0, \ldots, d/2$, k F_k^d, $x \in \{0,1\}^{2^d}$

$\log \mathcal{F}(x) \leq k$. k

NAND x $\mathcal{F}(x)$.

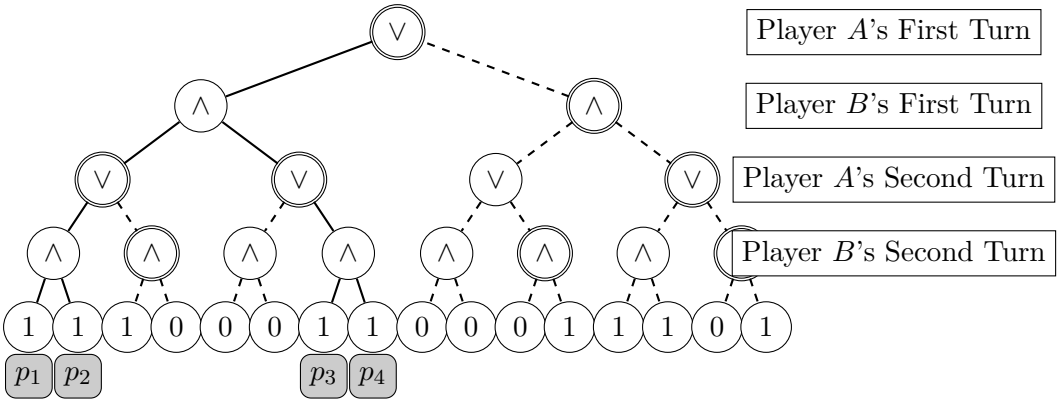


Figure 5: Depiction of a depth-4 NAND-tree as a two-player game. Let x be the input to NAND_4 shown in the figure. This instance is A-winnable, and $\mathcal{P}_A(x)$ consists of the paths $\{p_1, p_2, p_3, p_4\}$, shown using solid lines. Fault nodes are those with double circles. Each path in $\mathcal{P}_A(x)$ encounters two faults at nodes where Player A makes decisions. Therefore, $\mathcal{F}_A(x) = 4$.

$G_{\mathrm{NAND}_d}(x)$ $G'_{\mathrm{NAND}_d}(x)$ $\mathcal{F}(x)$:

[1] We have actually used the more refined definition of k-fault from [19].

Lemma 22. *For any $x \in \{0,1\}^{2^d}$, if d is even, then we have $R_{s,t}(G_{\mathrm{NAND}_d}(x)) \leq \mathcal{F}_A(x)$ and $R_{s',t'}(G'_{\mathrm{NAND}_d}(x)) \leq \mathcal{F}_B(x)$, while if d is odd, we have $R_{s,t}(G_{\mathrm{NAND}_d}(x)) \leq 2\mathcal{F}_A(x)$ and $R_{s',t'}(G'_{\mathrm{NAND}_d}(x)) \leq 2\mathcal{F}_B(x)$.*

[illegible]

Corollary 23. *The span program P_{G_ϕ} for $\phi = \mathrm{NAND}_d$ decides $\mathrm{EVAL}_{\mathrm{NAND}_d}$ restricted to the domain X in $O(\max_{x\in X} \mathcal{F}(x))$ queries. In particular, it decides k-fault trees (on domain F_k^d) in $O(2^k)$ queries.*

Proof of Theorem 20. Theorem 20 is now an immediate consequence of Corollary 23, with k set to $\log(d)$, along with the fact from [27] that the classical query complexity of evaluating such formulas is $\Omega(d^{\log\log(d)})$. □

[illegible]

Claim 24. *Let two graphs G_1 and G_2 each have nodes s and t and let $x^1 \in \{0,1\}^{E(G_1)}$ and $x^2 \in \{0,1\}^{E(G_2)}$. Suppose we create a new graph G by identifying the s nodes and the t nodes (i.e. connecting the graphs in parallel), then*

$$C_{s,t}(G(x^1,x^2)) = C_{s,t}(G_1(x^1)) + C_{s,t}(G_2(x^2)) \tag{29}$$

If we create a new graph G by identifying the t node of G_1 with the s node of G_2 and relabeling this node $v \notin \{s,t\}$ (i.e. connecting the graphs in series), then

$$C_{s,t}(G(x^1,x^2)) = \min\{C_{s,t}(G_1(x^1)), C_{s,t}(G_2(x^2))\}. \tag{30}$$

Proof of Theorem 21. Using Corollary 23, our analysis shows that $st\text{-CONN}_{G_{\mathrm{NAND}_d},F_1^d}$ can be decided in $O(1)$ queries.

We apply Eq. (8) to compare to the analysis of [4]. We must characterize the quantity $\max\{C_{s,t}(G_{\mathrm{NAND}_d}(x)) : x \in F_1^d, \text{and } s,t \text{ are not connected}\}$, since we already have

$$\max_{x\in F_1^d : s,t \text{ are connected}} R_{s,t}(G(x)) = O(1). \tag{31}$$

We now prove that for *every* $x \in \{0,1\}^{2^d}$ such that $\mathrm{NAND}_d(x) = 0$, $C_{s,t}(G_{\mathrm{NAND}_d}(x)) = 2^{\lfloor d/2 \rfloor}$. Thus, for any promise D on the input, as long as there exists some $x \in D$ such that $\mathrm{NAND}_d(x) = 0$, we have $\max_{x\in D:\mathrm{NAND}_d(x)=0} C_{s,t}(G_{\mathrm{NAND}_d}(x)) = 2^{\lfloor d/2 \rfloor}$. Intuitively, this is because every st-cut on any subgraph of G_{NAND_d} cuts across $2^{\lfloor d/2 \rfloor}$ edges of G_{NAND_d}.

The proof is by induction on d. For the base even case, $d = 0$ and G_{NAND_0} is a single edge connecting s and t. The only input $x \in \{0,1\}^{2^0}$ such that $\mathrm{NAND}_d(x) = 0$ is $x = 0$, in which case, the st-cut is $\kappa(s) = 1$ and $\kappa(t) = 0$, so the cut is across the unique edge in G_{NAND_0}, so $C_{s,t}(G_{\mathrm{NAND}_0}(x)) = 1$.

For the induction step, we treat even and odd separately. Suppose $d > 0$ is odd. Then $\mathrm{NAND}_d(x) = \mathrm{NAND}_{d-1}(x^0) \wedge \mathrm{NAND}_{d-1}(x^1)$, where $x^0 = (x_1, \ldots, x_{2^{d-1}})$ and $x^1 = (x_{2^{d-1}+1}, \ldots, x_{2^d})$. Thus $G_{\mathrm{NAND}_d}(x)$ involves composing two graphs $G_{\mathrm{NAND}_{d-1}}(x^0)$ and $G_{\mathrm{NAND}_{d-1}}(x^1)$ in series. Since we are assuming s and t are not connected in $G_{\mathrm{NAND}_d}(x)$, at least one of $G_{\mathrm{NAND}_{d-1}}(x^0)$ and $G_{\mathrm{NAND}_{d-1}}(x^1)$ must not be connected. Without loss of generality, suppose $G_{\mathrm{NAND}_{d-1}}(x^0)$ is not connected and $C_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^0)) \leq C_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^1))$. By induction, $C_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^0)) = 2^{(d-1)/2}$. Thus using Eq. (30) in Claim 24,

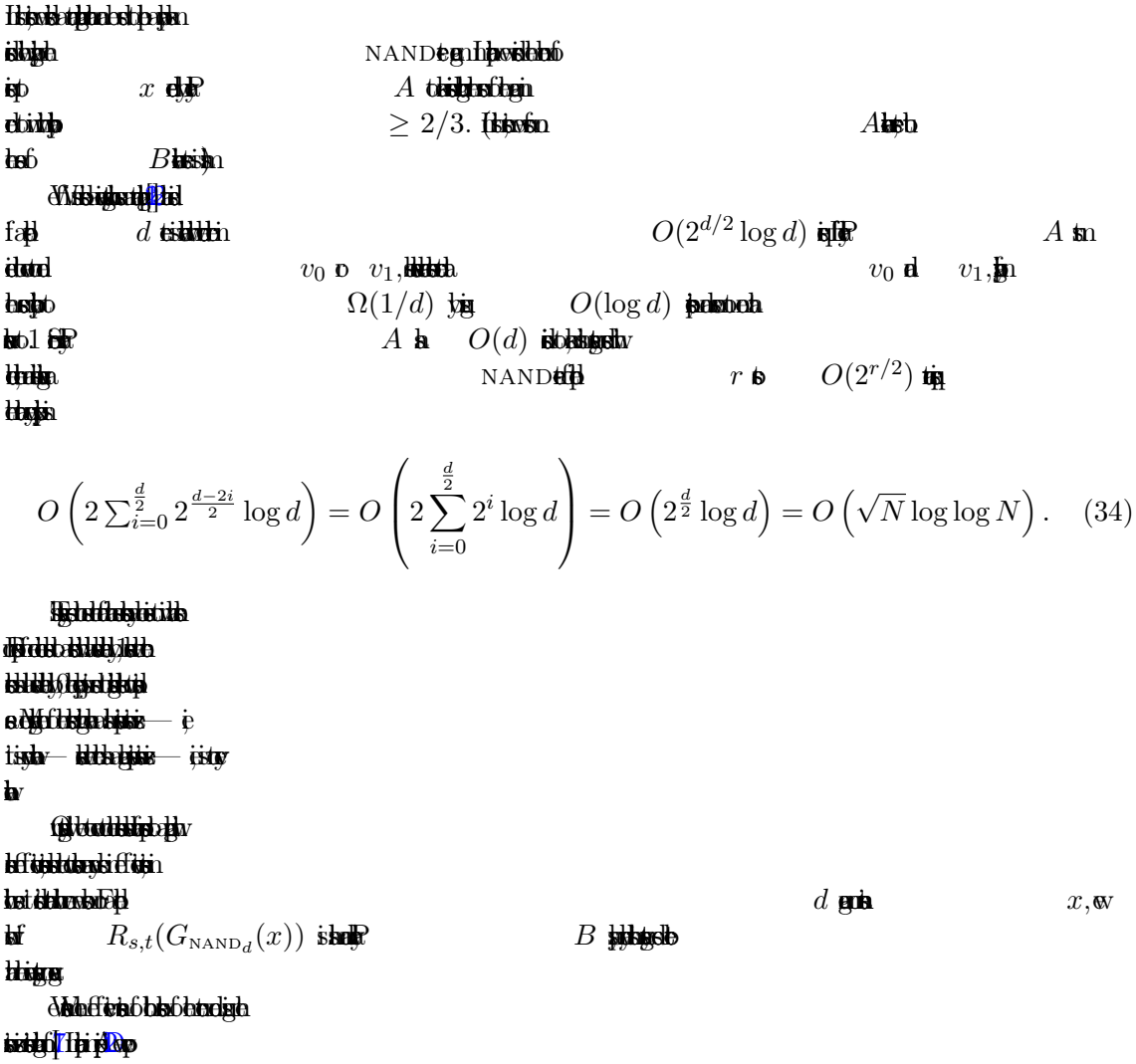
$$C_{s,t}(G_{\mathrm{NAND}_d}(x)) = 2^{(d-1)/2} = 2^{\lfloor d/2 \rfloor}. \tag{32}$$

Now suppose $d > 0$ is even. Then $\text{NAND}_d(x) = \text{NAND}_{d-1}(x^0) \vee \text{NAND}_{d-1}(x^1)$. Thus $G_{\text{NAND}_d}(x)$ involves composing two graphs $G_{\text{NAND}_{d-1}}(x^0)$ and $G_{\text{NAND}_{d-1}}(x^1)$ in parallel. Since we are assuming s and t are not connected in $G_{\text{NAND}_d}(x)$, both of $G_{\text{NAND}_{d-1}}(x^0)$ and $G_{\text{NAND}_{d-1}}(x^1)$ must not be connected, and so by induction, we have $C_{s,t}(G_{\text{NAND}_{d-1}}(x^0)) = C_{s,t}(G_{\text{NAND}_{d-1}}(x^1)) = 2^{\lfloor (d-1)/2 \rfloor} = 2^{d/2-1}$, since d is even. Thus using Eq. (29) in Claim 24,

$$C_{s,t}(G_{\text{NAND}_d}(x)) = 2^{d/2-1} + 2^{d/2-1} = 2^{d/2} = 2^{\lfloor d/2 \rfloor}. \tag{33}$$

Therefore, using Eq. (8), we have that the analysis of [4] for d-depth NAND-trees with inputs in F_1^d gives a query complexity of $O(\sqrt{2^{\lfloor d/2 \rfloor}}) = O(N^{1/4})$, where $N = 2^d$ is the number of input variables. Comparing with our analysis, which gives a query complexity of $O(1)$, we see there is a polynomial to constant improvement. □

5.2 Winning the NAND-tree Game



$$O\left(2\sum_{i=0}^{\frac{d}{2}} 2^{\frac{d-2i}{2}} \log d\right) = O\left(2\sum_{i=0}^{\frac{d}{2}} 2^{i} \log d\right) = O\left(2^{\frac{d}{2}} \log d\right) = O\left(\sqrt{N} \log\log N\right). \tag{34}$$

Lemma 25 (Est Algorithm)**.** *Let ϕ be an* AND-OR *formula with constant fan-in l, $\vee$-depth $d_\vee$ and $\wedge$-depth $d_\wedge$. Then the quantum query complexity of estimating $R_{s,t}(G_\phi(x))$ (resp. $R_{s,t}(G'_\phi(x))$) to relative accuracy ϵ is $\widetilde{\text{O}}\left(\frac{1}{\epsilon^{3/2}}\sqrt{R_{s,t}(G_\phi(x))l^{d_\vee}}\right)$ (resp. $\widetilde{\text{O}}\left(\frac{1}{\epsilon^{3/2}}\sqrt{R_{s,t}(G'_\phi(x))l^{d_\wedge}}\right)$).*

[4] A. Belovs. Learning-graph-based quantum algorithm for k-distinctness. In *Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science (FOCS 2012)*, pages 207–216. IEEE Computer Society, 2012. doi:10.1109/FOCS.2012.18.

[5] A. Belovs. Span programs for functions with constant-sized 1-certificates. In *Proceedings of the 44th Symposium on Theory of Computing (STOC 2012)*, pages 77–84. ACM, 2012. doi:10.1145/2213977.2213986.

[6] A. Belovs. Span-program-based quantum algorithm for the rank problem. *arXiv:1103.0842*, 2011.

[7] A. Belovs. Quantum algorithms for learning symmetric juntas via the adversary bound. In *Proceedings of the 29th Conference on Computational Complexity (CCC 2014)*, pages 22–31. IEEE Computer Society, 2014. doi:10.1109/CCC.2014.11.

[8] A. Belovs. Quantum walks and electrical networks. *arXiv:1302.3143*, 2013.

[9] A. Belovs. Learning-graph-based quantum algorithm for k-distinctness. *arXiv:1610.00581*, 2016.

[10] A. Belovs and R. Špalek. Adversary lower bound for the k-sum problem. In *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science (ITCS 2013)*, pages 323–328. ACM, 2013. doi:10.1145/2422436.2422475.

[11] A. Belovs and R. Špalek. Adversary lower bounds for detecting subgraphs. *Computational Complexity*, 22(4):811–871, 2013. doi:10.1007/s00037-013-0072-z.

[12] A. Belovs and T. Yolcu. On the minimum query complexity of the Forrelation problem. *Journal of the London Mathematical Society*, 2017. doi:10.1112/jlms.12060.

[13] P. G. Doyle and J. L. Snell. *Random Walks and Electrical Networks*, volume 22 of *The Carus Mathematical Monographs*. Mathematical Association of America, 1984.

[14] R. J. Duffin. Basic properties of discrete analytic functions. *Journal of Mathematical Analysis and Applications*, 23(2):335–363, 1968. doi:10.1016/0022-247X(68)90078-4.

[15] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser. Limit on the speed of quantum computation in determining parity. *Physical Review Letters*, 81(24):5442–5444, 1998. doi:10.1103/PhysRevLett.81.5442.

[16] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser. Quantum mechanical Hamiltonian models of Turing machines. *Journal of Statistical Physics*, 110(3–6):1013–1028, 2003. doi:10.1023/A:1022145429681.

[17] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser. Quantum mechanical Hamiltonian models of Turing machines that dissipate no energy. *Physical Review Letters*, 85(10):2005–2008, 2000. doi:10.1103/PhysRevLett.85.2005.

[18] L. K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the 28th annual ACM Symposium on Theory of computing (STOC 1996)*, pages 212–219. ACM, 1996. doi:10.1145/237814.237866.

[19] S. Jukna. *Boolean Function Complexity: Advances and Frontiers*, volume 27 of *Algorithms and Combinatorics*. Springer, 2012. doi:10.1007/978-3-642-24508-4.

[20] S. Jukna and G. Schnitger. On the optimality of Boolean functions. *Computational Complexity*, 24(1):177–217, 2015. doi:10.1007/s00037-013-0080-z.

[21] S. Lee, R. Mittal, and B. W. Reichardt. Span-program-based quantum algorithm for evaluating formulas. In *Proceedings of the 43rd International Colloquium on Automata, Languages, and Programming (ICALP 2016)*, pages 132:1–132:15. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2016. doi:10.4230/LIPIcs.ICALP.2016.132. arXiv:1507.00432.

[22] M. Saks. Randomization and derandomization in space-bounded computation. In *Proceedings of the IEEE 8th Annual Conference on Structure in Complexity Theory*, pages 128–149. IEEE Computer Society, 1993. doi:10.1109/SCT.1993.336538.

[23] M. Szegedy. Quantum speed-up of Markov chain based algorithms. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2004)*, pages 32–41. IEEE Computer Society, 2004. doi:10.1109/FOCS.2004.53.

[24] A. C. Yao. Lower bounds by probabilistic arguments. In *Proceedings of the 24th Annual Symposium on Foundations of Computer Science (SFCS 1983)*, pages 420–428. IEEE Computer Society, 1983. doi:10.1109/SFCS.1983.30.

Proceedings of the 52nd Annual IEEE Symposium on Foundations of Computer Science (FOCS 2011),

Proceedings of the 50th IEEE Symposium on Foundations of Computer Science (FOCS 2009),

arXiv:quant-ph/0904.2759.

Electronic Colloquium on Computational Complexity (ECCC),

Proceedings of the 22nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2011),

Theory of Computing,

Proceedings of the 27th Annual Symposium on Foundations of Computer Science (FOCS 1986),

Proceedings of the 11th Annual ACM Symposium on Theory of Computing (STOC 1979),

Proceedings of the 3rd Innovations in Theoretical Computer Science Conference (ITCS 2012),

A Analysis of the Span Program for *st*-Connectivity

$P_{G,c}$

G.

Definition 28 (Circulation)**.** *A* circulation *on a graph G is a function $\theta : \overrightarrow{E}(G) \to \mathbb{R}$ such that:*

1. *For all $(u, v, \lambda) \in \overrightarrow{E}(G)$, $\theta(u, v, \lambda) = -\theta(v, u, \lambda)$;*
2. *for all $u \in V(G)$, $\sum_{v,\lambda:(u,v\lambda)\in\overrightarrow{E}(G)} \theta(u, v, \lambda) = 0$.*

Claim 29. *Let θ be a unit st-flow in some multigraph G. We can consider the corresponding vector $|\theta\rangle = \sum_{(u,v,\lambda)\in\overrightarrow{E}(G)} \theta(u, v, \lambda)|u, v, \lambda\rangle$. Then $|\theta\rangle$ can be written as a linear combination*

of vectors corresponding to self-avoiding st-paths and cycles that are edge-disjoint from these paths.

Let σ be a circulation on G. Then $|\sigma\rangle$ can be written as a linear combination of cycles in G. Furthermore, $|\sigma\rangle$ can be written as a linear combination of cycles such that each cycle goes around a face of G.

[illegible] *st*-flow

Claim 30. *Fix a span program $P_{G,c}$ as in (6). Call $|w\rangle \in H$ a positive witness in $P_{G,c}$ if $A|w\rangle = \tau$ (note that such a $|w\rangle$ is not necessarily a positive witness for any particular input x). Then if θ is a unit st-flow in G, $\frac{1}{2}\sum_{(u,v,\lambda)\in\vec{E}(G)} \frac{\theta(u,v,\lambda)}{\sqrt{c(\{u,v\},\lambda)}}|u,v,\lambda\rangle$ is a positive witness in $P_{G,c}$, and furthermore, if $|w\rangle$ is a positive witness in $P_{G,c}$, then $\theta(u,v,\lambda) = \sqrt{c(\{u,v\},\lambda)}(\langle w|u,v,\lambda\rangle - \langle w|v,u,\lambda\rangle)$ is a unit st-flow in G.*

Proof. The proof is a straightforward calculation. Let θ be a unit st-flow on G. Then

$$\begin{aligned}
&A\left(\frac{1}{2}\sum_{(u,v,\lambda)\in\vec{E}(G)} \frac{\theta(u,v,\lambda)}{\sqrt{c(\{u,v\},\lambda)}}|u,v,\lambda\rangle\right) \\
&= \frac{1}{2}\sum_{(u,v,\lambda)\in\vec{E}(G)} \theta(u,v,\lambda)(|u\rangle - |v\rangle) \\
&= \frac{1}{2}\sum_{u\in V(G)}\left(\sum_{v,\lambda:(u,v,\lambda)\in\vec{E}(G)} \theta(u,v,\lambda)\right)|u\rangle + \frac{1}{2}\sum_{v\in V(G)}\left(\sum_{u,\lambda:(v,u,\lambda)\in\vec{E}(G)} \theta(v,u,\lambda)\right)|v\rangle \\
&= \frac{1}{2}(|s\rangle - |t\rangle) + \frac{1}{2}(|s\rangle - |t\rangle) = \tau. \qquad (35)
\end{aligned}$$

Above we have used that $\theta(u,v,\lambda) = -\theta(v,u,\lambda)$, and $\sum_{v,\lambda:(u,v,\lambda)\in\vec{E}(G)} \theta(u,v,\lambda) = 0$ when $u \notin \{s,t\}$, 1 when $u = s$, and -1 when $u = t$.

To prove the second half of the claim, let $|w\rangle$ be such that $A|w\rangle = \tau$, and define $\theta(u,v,\lambda) = \sqrt{c(\{u,v\},\lambda)}(\langle w|u,v,\lambda\rangle - \langle w|v,u,\lambda\rangle)$. We immediately see that $\theta(u,v,\lambda) = -\theta(v,u,\lambda)$ for all (u,v,λ). Furthermore, we have:

$$\begin{aligned}
|s\rangle - |t\rangle = A|w\rangle &= \sum_{u\in V(G)}\left(\sum_{(u,v,\lambda)\in\vec{E}(G)} \sqrt{c(\{u,v\},\lambda)}\langle u,v,\lambda|w\rangle\right)|u\rangle \\
&\quad - \sum_{v\in V(G)}\left(\sum_{(u,v,\lambda)\in\vec{E}(G)} \sqrt{c(\{u,v\},\lambda)}\langle u,v,\lambda|w\rangle\right)|v\rangle \\
&= \sum_{u\in V(G)}\left(\sum_{(u,v,\lambda)\in\vec{E}(G)} \sqrt{c(\{u,v\},\lambda)}(\langle u,v,\lambda|w\rangle - \langle v,u,\lambda|w\rangle)\right)|u\rangle \\
&= \sum_{u\in V(G)}\left(\sum_{(u,v,\lambda)\in\vec{E}(G)} \theta(u,v,\lambda)\right)|u\rangle. \qquad (36)
\end{aligned}$$

Thus, for all $u \in V(G(x))\backslash\{s,t\}$, $\sum_{(u,v,\lambda)\in\vec{E}(G)} \theta(u,v,\lambda) = 0$, and $\sum_{(s,v,\lambda)\in\vec{E}(G)} \theta(s,v,\lambda) = \sum_{(v,t,\lambda)\in\vec{E}(G)} \theta(v,t,\lambda) = 1$. Thus, θ is a unit st-flow on G. □

[illegible] $s't'$-flow

Claim 31. *For a planar graph G, fix a span program $P_{G,c}$ as in (6). Call a linear function $\omega : V(G) \to \mathbb{R}$ a negative witness if $\omega\tau = 1$. Then $\theta((u,v,\lambda)^\dagger) = \omega(u) - \omega(v)$ is a unit $s't'$-flow on G', and furthermore, for every $s't'$-flow θ on G' there is a negative witness ω such that $\theta((u,v,\lambda)^\dagger) = \omega(u) - \omega(v)$ for all $(u,v,\lambda) \in \overrightarrow{E}(G)$.*

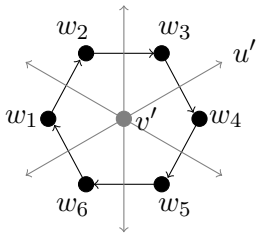


Figure 6: The duality between a cycle and a star.

Proof. When we consider the edges of G as directed edges, we assign edge directions to the dual by orienting each dual edge $\pi/2$ radians counter-clockwise from the primal edge.

Note that without loss of generality, if ω is a negative witness, we can assume $\omega(s) = 1$ and $\omega(t) = 0$. This is because $\|\omega A\|$ and $\left\|\omega A \Pi_{H(x)}\right\|$ are invariant under affine transformations of ω.

We first show that if ω is a negative witness in $P_{G,c}$, then $\theta : \overrightarrow{E}(G') \to \mathbb{R}$ defined $\theta((u,v,\lambda)^\dagger) = \omega(u) - \omega(v)$ is a unit $s't'$-flow on G'. To begin with, we will define $\theta'((u,v,\lambda)^\dagger) = \omega(u) - \omega(v)$ on $\overrightarrow{E}(\overline{G}^\dagger)$, so θ' agrees with θ everywhere θ is defined, and in addition, $\theta'(s',t',\emptyset) = \theta'((s,t,\emptyset)^\dagger) = \omega(s) - \omega(t) = 1$, and $\theta'(t',s',\emptyset) = -1$. Then clearly we have $\theta'(u',v',\lambda) = -\theta'(v',u',\lambda)$ for all $(\{u',v'\},\lambda) \in E(\overline{G}^\dagger)$.

Next, every $v' \in V(\overline{G}^\dagger)$ corresponds to a face $f_{v'}$ of $\overline{G}$, and the edges coming out of v' are dual to edges going clockwise around the face $f_{v'}$ (see Figure 6). If $(w_1,w_2,\lambda_1),\ldots,(w_k,w_{k+1},\lambda_k)$, for $w_{k+1} = w_1$, are the directed edges going clockwise around $f_{v'}$, then we have:

$$0 = \sum_{i=1}^{k} (\omega(w_i) - \omega(w_{i+1})) = \sum_{i=1}^{k} \theta'((w_i,w_{i+1},\lambda_i)^\dagger) = \sum_{\substack{u',\lambda: \\ (\{v',u'\},\lambda)\in E(\overline{G}')}} \theta'(v',u',\lambda). \tag{37}$$

Thus, θ' is a circulation. Then, since $\theta'(s',t',\emptyset) = 1$, if we remove the flow on this edge, which recovers θ, we get a unit $s't'$-flow on G'.

Next we will show that if θ is a unit $s't'$-flow on G', then there exists a negative witness ω in $P_{G,c}$ such that for all $(u,v,\lambda) \in \overrightarrow{E}(G)$, $\theta((u,v,\lambda)^\dagger) = \omega(u) - \omega(v)$.

Define θ' to be the circulation on $\overline{G}^\dagger$ obtained from defining $\theta'(u',v',\lambda) = \theta(u',v',\lambda)$ for all $(u',v',\lambda) \in \overrightarrow{E}(G')$, and $\theta'(s',t',\emptyset) = -\theta'(t',s',\emptyset) = 1$. Then if we define $|\theta'\rangle = \sum_{(u,v,\lambda)\in\overrightarrow{E}(\overline{G}^\dagger)} \theta'(u,v,\lambda)|u,v,\lambda\rangle$, we can express $|\theta'\rangle$ as a linear combination of cycles around the faces of $\overline{G}^\dagger$, $|\theta'\rangle = \sum_{f\in F(\overline{G}^\dagger)} \alpha_f |\overrightarrow{C}_f\rangle + \sum_{f\in F(\overline{G}^\dagger)} \alpha'_f |\overleftarrow{C}_f\rangle$, where if $w_{k+1} = w_1$ and $(w_1,w_2,\lambda_1),\ldots,(w_k,w_{k+1},\lambda_k)$ is a clockwise cycle around f, $|\overrightarrow{C}_f\rangle = \sum_{i=1}^{k} |w_i,w_{i+1},\lambda_i\rangle$ is the clockwise cycle around the face f, and $|\overleftarrow{C}_f\rangle = \sum_{i=1}^{k} |w_{i+1},w_i,\lambda_i\rangle$ is the counter-clockwise cycle around f. There is a one-to-one correspondance between vertices in $V(\overline{G}) = V(G)$ and faces in $F(\overline{G}^\dagger)$, so we can define $\omega : V(G) \to \mathbb{R}$ by $\omega(v_f) = \frac{1}{2}(\alpha_f - \alpha'_f)$.

We claim that for all $(u,v,\lambda) \in \overrightarrow{E}(\overline{G})$, $\omega(u) - \omega(v) = \theta'((u,v,\lambda)^\dagger)$. Let (u',v',λ) be any edge in $\overrightarrow{E}(\overline{G}^\dagger)$. This edge is part of a clockwise cycle around one face in $\overline{G}^\dagger$, call it f,

and a counter clockwise cycle around one face in $\overline{G}^\dagger$, call it g. Since these are the only two faces containing the edge (u', v', λ), we must have $\theta'(u', v', \lambda) = \langle u', v', \lambda | \theta' \rangle = \alpha_f + \alpha'_g$. Since $\theta'(u', v', \lambda) = -\theta'(v', u', \lambda)$, we have $\alpha_f + \alpha'_g = -\alpha'_f - \alpha_g$. Thus:

$$\omega(v_f) - \omega(v_g) = \frac{1}{2}\left(\alpha_f - \alpha'_f - \alpha_g + \alpha'_g\right) = \frac{1}{2}\left(\theta'(u', v', \lambda) - \theta'(v', u', \lambda)\right) = \theta'((v_f, v_g, d)^\dagger). \tag{38}$$

In particular, this means that $\omega(s) - \omega(t) = \theta'((s, t, \emptyset)^\dagger) = \theta'(s', t', \emptyset) = 1$, so ω is a negative witness, and for all $(u, v, \lambda) \in \overrightarrow{E}(G)$, $\omega(u) - \omega(v) = \theta((u, v, \lambda)^\dagger)$. □

Lemma 11. *Let G be a planar multigraph with $s, t \in V(G)$ such that $G \cup \{\{s, t\}\}$ is also planar, and let c be a weight function on $E(G)$. Let $x \in \{0, 1\}^{E(G)}$. Then $w_+(x, P_{G,c}) = \frac{1}{2} R_{s,t}(G(x), c)$ and $w_-(x, P_{G,c}) = 2R_{s',t'}(G'(x), c')$.*

Proof. If x is a 1-instance, s and t are connected in $G(x)$, so there exists a unit st-flow on $G(x)$, which is a unit st-flow on G that is supported only on $\overrightarrow{E}(G(x))$. Let θ be the flow on $G(x)$ such that $R_{s,t}(G(x), c) = \sum_{(\{u,v\},\lambda) \in E(G(x))} \frac{\theta(u,v,\lambda)^2}{c(\{u,v\},\lambda)}$. By Claim 30, $|w\rangle = \frac{1}{2} \sum_{(u,v,\lambda)} \frac{\theta(u,v,\lambda)}{\sqrt{c(\{u,v\},\lambda)}} |u, v, \lambda\rangle$ is a positive witness in $P_{G,c}$, and since θ is supported on $\overrightarrow{E}(G(x))$, $|w\rangle \in H(x)$, and so $|w\rangle$ is a positive witness for x in $P_{G,c}$. Thus

$$w_+(x, P_{G,c}) \leq \||w\rangle\|^2 = \frac{1}{4} \sum_{(u,v,\lambda) \in \overrightarrow{E}(G(x))} c(\{u, v\}, \lambda)\theta(u, v, \lambda)^2 = \frac{1}{2} R_{s,t}(G(x), c). \tag{39}$$

On the other hand, let $|w\rangle$ be an optimal positive witness for x. By Claim 30, $\theta(u, v, \lambda) = \sqrt{c(\{u, v\}, \lambda)}(\langle u, v, \lambda | w\rangle - \langle v, u, \lambda | w\rangle)$ is a unit st-flow on G, and since $|w\rangle \in H(x)$, $\theta(u, v, \lambda)$ is only non-zero on $\overrightarrow{E}(G(x))$, so θ is a unit st-flow on $G(x)$. Thus,

$$\begin{aligned} R_{s,t}(G(x), c) &\leq \sum_{(\{u,v\},\lambda) \in E(G(x))} \frac{\theta(u, v, \lambda)^2}{c(\{u, v\}, \lambda)} = \frac{1}{2} \sum_{(u,v,\lambda) \in \overrightarrow{E}(G(x))} \left(\langle u, v, \lambda | w\rangle - \langle v, u, \lambda | w\rangle\right)^2 \\ &= \sum_{(u,v,\lambda) \in \overrightarrow{E}(G(x))} \langle u, v, \lambda | w\rangle^2 - \sum_{(u,v,\lambda) \in \overrightarrow{E}(G(x))} \langle u, v, \lambda | w\rangle \langle v, u, \lambda | w\rangle \leq 2 \||w\rangle\|^2 \end{aligned} \tag{40}$$

where the last inequality is by Cauchy-Schwarz. Thus, $w_+(x, P_{G,c}) = \frac{1}{2} R_{s,t}(G(x))$.

Now we prove that $w_-(x, P_{G,c}) = 2R_{s',t'}(G'(x), c')$. Let $x \in \{0, 1\}^{E(G)}$ be such that s and t are not connected in $G(x)$. Fix an optimal negative witness ω for x. By Claim 31 the linear function $\theta : \overrightarrow{E}(G') \to \mathbb{R}$ defined by $\theta((u, v, \lambda)^\dagger) = \omega(u) - \omega(v)$ is a unit $s't'$ -flow on G'. Since ω is a negative witness for x, we also have:

$$\begin{aligned} 0 = \left\|\omega A \Pi_{H(x)}\right\|^2 &= \sum_{(u,v,\lambda) \in \overrightarrow{E}(G(x))} c(\{u, v\}, \lambda)(\omega(u) - \omega(v))^2 \\ &= \sum_{(u,v,\lambda) \in \overrightarrow{E}(G(x))} c(\{u, v\}, \lambda)\theta((u, v, \lambda)^\dagger)^2 \\ &= \sum_{(u',v',\lambda) \in \overrightarrow{E}(G') \setminus \overrightarrow{E}(G'(x))} \frac{\theta(u', v', \lambda)^2}{c'(\{u', v'\}, \lambda)}, \end{aligned} \tag{41}$$

since $(u,v,\lambda) \in \overrightarrow{E}(G(x))$ exactly when $(u,v,\lambda)^\dagger \notin \overrightarrow{E}(G'(x))$. So θ is only supported on $\overrightarrow{E}(G'(x))$, and so it is a unit $s't'$-flow on $G'(x)$. Thus

$$w_-(x, P_{G,c}) = \|\omega A\|^2 = \sum_{(u,v,\lambda)\in\overrightarrow{E}(G)} c(\{u,v\},\lambda)(\omega(u)-\omega(v))^2$$
$$= \sum_{(u',v',\lambda)\in\overrightarrow{E}(G'(x))} \frac{\theta(u',v',\lambda)^2}{c'(\{u',v'\},\lambda)} \geq 2R_{s',t'}(G'(x),c'). \qquad (42)$$

For the other direction, let θ be an $s't'$-flow in $G'(x)$ with minimal energy. By Claim 31, there is a negative witness ω such that $\theta((u,v,\lambda)^\dagger) = \omega(u) - \omega(v)$. Since θ is supported on edges $(u',v',\lambda) \in \overrightarrow{E}(G'(x))$, which are exactly those edges such that $(u',v',\lambda)^\dagger \notin \overrightarrow{E}(G(x))$, we have

$$0 = \sum_{\substack{(u,v,\lambda)\\ \in\overrightarrow{E}(G(x))}} c(\{u,v\},\lambda)\theta((u,v,\lambda)^\dagger)^2 = \sum_{\substack{(u,v,\lambda)\\ \in\overrightarrow{E}(G(x))}} c(\{u,v\},\lambda)(\omega(u)-\omega(v))^2 = \left\|\omega A \Pi_{H(x)}\right\|^2, \qquad (43)$$

so ω is a negative witness for x in $P_{G,c}$. Thus:

$$w_-(x, P_{G,c}) \leq \|\omega A\|^2 = \sum_{(u,v,\lambda)\in\overrightarrow{E}(G)} c(\{u,v\},\lambda)(\omega(u)-\omega(v))^2$$
$$= \sum_{(u',v',\lambda)\in\overrightarrow{E}(G'(x))} \frac{\theta(u',v',\lambda)^2}{c'(\{u',v'\},\lambda)} = 2R_{s',t'}(G'(x),c'), \qquad (44)$$

completing the proof. □

A.1 Time and Space Analysis of the Span Program Algorithm for *st*-Connectivity

[illegible] st-CONN$_G$ [illegible] G. [illegible] is $O(\max\{\log|E(G)|, \log|V(G)|\})$.

[illegible] Π_S [illegible] S. [illegible] $P = (H, U, A, \tau)$, [illegible] $(2\Pi_{H(x)} - I)(2\Pi_{\ker A} - I)$ [illegible] $|w_0\rangle = A^+\tau$, [illegible] $\Pi_{H(x)}$ [illegible] $H(x)$, [illegible] $\Pi_{\ker A}$ [illegible] A, [illegible] A^+ [illegible] A. [illegible] f [illegible] D, [illegible]

$$O\left(\sqrt{\max_{x\in D:f(x)=1} w_+(x) \times \max_{x\in D:f(x)=0} w_-(x)}\right).$$

[illegible] st [illegible] $P_{G,c}$ in (6), [illegible] $2\Pi_{H(x)} - I$ [illegible] $O(1)$ [illegible] x. [illegible] $2\Pi_{\ker A} - I$ [illegible] x, [illegible] G. [illegible] $2\Pi_{\ker A} - I$ [illegible] G, [illegible] G [illegible] G is [illegible] n [illegible] $O(\log n)$ [illegible]

[illegible] G [illegible] c, [illegible] *quantum walk step on G* [illegible] [illegible] $U_{G,c}$ [illegible] $u \in V(G)$:

$$U_{G,c} : |u\rangle|0\rangle \mapsto \frac{1}{\sqrt{\sum_{v,\lambda:(u,v,\lambda)\in\overrightarrow{E}(G)} c(\{u,v\},\lambda)}} \sum_{v,\lambda:(u,v,\lambda)\in\overrightarrow{E}(G)} \sqrt{c(\{u,v\},\lambda)}|u\rangle|u,v,\lambda\rangle. \tag{45}$$

Theorem 13. *Let $P_{G,c} = (H, U, A, \tau)$ be defined as in (6). Let $S_{G,c}$ be an upper bound on the time complexity of implementing $U_{G,c}$. If G has the property that $G \cup \{\{s,t\}\}$ is planar, then the time complexity of deciding st-CONN$_{G,D}$ is at most*

$$O\left(\min_c S_{G,c}\sqrt{\max_{x\in D: s,t \text{ are connected}} R_{s,t}(G(x),c) \times \max_{x\in D: s,t \text{ are not connected}} R_{s',t'}(G'(x),c')}\right). \tag{12}$$

[illegible]

Lemma 32. *Let A be defined as in (6). Let $S_{G,c}$ be an upper bound on the time complexity of implementing $U_{G,c}$. Then $2\Pi_{\ker A} - I$ can be implemented in time complexity $O(S_{G,c})$.*

Proof. This analysis follows [4] (see also [17]). Let

$$d(u) = \sum_{v,\lambda:(u,v,\lambda)\in\overrightarrow{E}(G)} c(\{u,v\},\lambda). \tag{46}$$

Define spaces Z and Y as follows.

$$Z = \text{span}\left\{|z_u\rangle := \sum_{v,\lambda:(u,v,\lambda)\in\overrightarrow{E}(G)} \frac{\sqrt{c(\{u,v\},\lambda)}}{\sqrt{2d(u)}} (|0,u,u,v,\lambda\rangle + |1,u,v,u,\lambda\rangle) : u \in V(G)\right\} \tag{47}$$

$$Y = \text{span}\left\{|y_{u,v,\lambda}\rangle := \frac{|0,u,u,v,\lambda\rangle - |1,v,u,v,\lambda\rangle}{\sqrt{2}} : (u,v,\lambda) \in \overrightarrow{E}(G)\right\} \tag{48}$$

Define isometries whose column-spaces are Z and Y respectively:

$$M_Z = \sum_{u\in V(G)} |z_u\rangle\langle u| \quad \text{and} \quad M_Y = \sum_{(u,v,\lambda)\in\overrightarrow{E}(G)} |y_{u,v,\lambda}\rangle\langle u,v,\lambda|. \tag{49}$$

Now we note that for any $(\{u,v\},\lambda) \in E(G)$, we have the following:

$$\langle z_u|y_{u,v,\lambda}\rangle = \frac{\sqrt{c(\{u,v\},\lambda)}}{2\sqrt{d(u)}}, \quad \text{and} \quad \langle z_v|y_{u,v,\lambda}\rangle = -\frac{\sqrt{c(\{u,v\},\lambda)}}{2\sqrt{d(v)}}. \tag{50}$$

Thus, we can calculate:

$$\begin{aligned} M_Z^\dagger M_Y &= \sum_{(u,v,\lambda)\in\overrightarrow{E}(G)} \left(\frac{|u\rangle}{2\sqrt{d(u)}} - \frac{|v\rangle}{2\sqrt{d(v)}}\right)\sqrt{c(\{u,v\},\lambda)}\langle u,v,\lambda| \\ &= \sum_{u'\in V(G)} \frac{|u'\rangle\langle u'|}{2\sqrt{d(u')}} \sum_{(u,v,\lambda)\in\overrightarrow{E}(G)} \sqrt{c(\{u,v\},\lambda)}(|u\rangle - |v\rangle)\langle u,v,\lambda| \\ &= \sum_{u'\in V(G)} \frac{|u'\rangle\langle u'|}{2\sqrt{d(u')}} A. \end{aligned} \tag{51}$$

Note that the rows of $M_Z^\dagger M_Y$ are non-zero multiples of the rows of A, so $\mathrm{row}(M_Z^\dagger M_Y) = \mathrm{row}(A)$, and thus $\ker(M_Z^\dagger M_Y) = \ker A$.

Define $W = (2\Pi_Z - I)(2\Pi_Y - I)$. We now claim that M_Y maps $\ker A$ to the (-1)-eigenspace of W, and $(\ker A)^\perp$ to the 1-eigenspace of W, so that $2\Pi_{\ker A} - I = M_Y^\dagger W M_Y$. To see this, note that if $|\psi\rangle \in \ker A$, then $|\psi\rangle \in \ker(M_Z^\dagger M_Y)$ so $M_Y|\psi\rangle \in \ker M_Z^\dagger = Z^\perp$. Thus $M_Y|\psi\rangle \in Y \cap Z^\perp$, which is in the (-1)-eigenspace of W.

Next, suppose $|\psi\rangle \in (\ker A)^\perp = (\ker(M_Z^\dagger M_Y))^\perp$, so since M_Y is an isometry, $M_Y|\psi\rangle \in (\ker M_Z)^\perp = \mathrm{row} M_Z = Z$. Thus $M_Y|\psi\rangle \in Y \cap Z$, which is in the 1-eigenspace of W.

Thus, we can implement $2\Pi_{\ker A} - I$ by $M_Y^\dagger W M_Y$. It only remains to argue that each of M_Y, $2\Pi_Z - I$ and $2\Pi_Y - I$ can be implemented in time complexity at most $O(S_{G,c})$.

We first show that we can implement the isometry M_Y, or rather a unitary U_Y that acts as $|0\rangle|0\rangle|u,v,\lambda\rangle \mapsto M_Y|u,v,\lambda\rangle = |y_{u,v,\lambda}\rangle$. First, use HX on the first qubit to perform the map:

$$|0\rangle|0\rangle|u,v,\lambda\rangle \mapsto |-\rangle|0\rangle|u,v,\lambda\rangle. \tag{52}$$

Conditioned on the value of the first register, copy either u or v into the second register to get:

$$\frac{1}{\sqrt{2}}(|0,u,u,v,\lambda\rangle - |1,v,u,v,\lambda\rangle) = |y_{u,v,\lambda}\rangle. \tag{53}$$

Thus, we can implement U_Y in the time it takes to write down a vertex of G, $O(\log|V(G)|)$, which is at most $O(S_{G,c})$. Using the ability to implement U_Y, we can implement $2\Pi_Y - I$ as $U_Y R_Y U_Y^\dagger$, where R_Y is the reflection that acts as the identity on computational basis states of the form $|0\rangle|0\rangle|u,v,\lambda\rangle$, and reflects computational basis states without this form.

Next, we implement a unitary U_Z that acts as $|0\rangle|u\rangle|0\rangle \mapsto M_Z|u\rangle = |z_u\rangle$. First, use the quantum walk step $U_{G,c}$, which can be implemented in time $S_{G,c}$, to perform:

$$|+\rangle|u\rangle|0\rangle \mapsto \frac{1}{2\sqrt{d(u)}} \sum_{v,\lambda:(u,v,\lambda)\in\vec{E}(G)} \sqrt{c(\{u,v\},\lambda)}(|0\rangle + |1\rangle)|u\rangle|u,v,\lambda\rangle. \tag{54}$$

Conditioned on the bit in the first register, swap the third and fourth registers, to get:

$$\frac{1}{2\sqrt{d(u)}} \sum_{v,\lambda:(u,v,\lambda)\in\vec{E}(G)} \sqrt{c(\{u,v\},\lambda)}(|0\rangle|u\rangle|u,v,\lambda\rangle + |1\rangle|u\rangle|v,u,\lambda\rangle) = |z_u\rangle. \tag{55}$$

The total cost of implementing U_Z is $O(S_{G,c} + \log|V(G)|) = O(S_{G,c})$. Thus, we can implement $2\Pi_Z - I$ in $O(S_{G,c})$ quantum gates. □

Lemma 33. *Let A and τ be defined as in (6). Let $S_{G,c}$ be an upper bound on the complexity of implementing $U_{G,c}$. Then the initial state of the algorithm, $\frac{|w_0\rangle}{\||w_0\rangle\|}$ where $|w_0\rangle = A^+\tau$, can be approximated in time $O(S_{G,c})$.*

Proof. Without loss of generality, we can assume that G includes the edge $(\{s,t\},\emptyset)$ (we can simply not include it in any subgraph). Furthermore, we set $c(\{s,t\},\emptyset) = 1/r$, for some positive r to be specified later, so that $A|s,t,\emptyset\rangle = r^{-1/2}\tau$. This has no effect on other edges in G. Note that

$$\Pi_{(\ker A)^\perp}|s,t,\emptyset\rangle = A^+A|s,t,\emptyset\rangle = r^{-1/2}A^+\tau = r^{-1/2}|w_0\rangle, \tag{56}$$

so

$$|s,t,\emptyset\rangle = r^{-1/2}|w_0\rangle + |w_0^{\perp}\rangle \tag{57}$$

for some $|w_0^{\perp}\rangle \in \ker A$. Thus, constant precision phase estimation on $2\Pi_{\ker A} - I$ maps $|s,t,\emptyset\rangle$ to

$$r^{-1/2}|0\rangle|w_0\rangle + |1\rangle|w_0^{\perp}\rangle. \tag{58}$$

Using quantum amplitude amplification [7], we can amplify the amplitude on the $|0\rangle|w_0\rangle$ part of this arbitrarily close to 1 using a number of calls to $2\Pi_{\ker A} - I$ proportional to $\left\| r^{-1/2}|w_0\rangle \right\|^{-1}$.

In fact, it is straightforward to show that for any $|\mu\rangle \in \mathrm{row}A$, the vector $|\nu\rangle$ with smallest norm that satisfies $A|\nu\rangle = |\mu\rangle$, is $A^{+}|\mu\rangle$ [17]. Using this fact along with Claim 30 and Definition 9, we have $\||w_0\rangle\|^2 = R_{s,t}(G,c)$.

Let $R = R_{s,t}(G \setminus \{(\{s,t\},\emptyset)\},c)$ be the effective resistance of G without the edge $(s,t,\emptyset)$. Now we can think of $(\{s,t\},\emptyset)$ and $G \setminus \{(\{s,t\},\emptyset)\}$ as two graphs in parallel, so using Claim 5, we have

$$\||w_0\rangle\|^2 = \frac{1}{1/R + 1/r}. \tag{59}$$

Setting $r = R$, we have $\||w_0\rangle\|^2 = R/2$ and $\|r^{-1/2}|w_0\rangle\|^{-1} = O(1)$. Thus, using $O(1)$ calls to $2\Pi_{\ker A} - I$, we can approximate the initial state $|w_0\rangle$. □

$U_{G,c}$, is $O(\max\{\log|E(G)|, \log|V(G)|\})$. U_Y $4|V(G)|^2|E(G)|$, $O(\max\{\log|E(G)|, \log|V(G)|\})$ $U_{G,c}$ $S'_{G,c}$ $(2\Pi_{H(x)} - I)(2\Pi_{\ker A} - I)$

$$O\left(\min_c \sqrt{\max_{x\in D:\phi(x)=1} R_{s,t}(G_\phi(x),c) \times \max_{x\in D:\phi(x)=0} R_{s',t'}(G'_\phi(x),c')}\right) = O(|E(G)|). \tag{60}$$

$O(\log(|E(G)|)$

B Formula Evaluation and *st*-Connectivity

ϕ, G_ϕ. G_ϕ.

Definition 34 (G_ϕ)**.** *If $\phi = x_i$ is a single-variable formula, then $V(G_\phi) = \{s,t\}$ and $E(G_\phi) = \{(\{s,t\},x_i)\}$.*

If $\phi = \phi_1 \wedge \cdots \wedge \phi_l$, then define $V(G_\phi) = \{(i,v) : i \in [l], v \in V(G_{\phi_i}) \setminus \{s,t\}\} \cup \{s, s_2, \ldots, s_l, t\}$ and, letting $s_1 = s$ and $s_{l+1} = t$, define:

$$\begin{aligned} E(G_\phi) = &\{(\{(i,u),(i,v)\},x_j) : i \in [l], u,v \in V(G_{\phi_i}) \setminus \{s,t\}, (\{u,v\},x_j) \in E(G_{\phi_i})\} \\ &\cup \{(\{(i,u),s_i\},x_j) : i \in [l], u \in V(G_{\phi_i}), (\{s,u\},x_j) \in E(G_{\phi_i})\} \\ &\cup \{(\{(i,u),s_{i+1}\},x_j) : i \in [l], u \in V(G_{\phi_i}), (\{t,u\},x_j) \in E(G_{\phi_i})\}. \end{aligned} \tag{61}$$

If $\phi = \phi_1 \vee \cdots \vee \phi_l$ *define* $V(G_\phi) = \{(i,v) : i \in [l], v \in V(G_{\phi_i}) \setminus \{s,t\}\} \cup \{s,t\}$ *and*

$$\begin{aligned} E(G_\phi) =& \{(\{(i,u),(i,v)\}, x_j) : i \in [l], u,v \in V(G_{\phi_i}) \setminus \{s,t\}, (\{u,v\}, x_j) \in E(G_{\phi_i})\} \\ &\cup \{(\{(i,u),s\}, x_j) : i \in [l], u \in V(G_{\phi_i}), (\{u,s\}, x_j) \in E(G_{\phi_i})\} \\ &\cup \{(\{(i,u),t\}, x_j) : i \in [l], u \in V(G_{\phi_i}), (\{u,t\}, x_j) \in E(G_{\phi_i})\}. \end{aligned} \tag{62}$$

Lemma 35. *For an* AND-OR *formula* ϕ *on* $\{0,1\}^N$*, define* ϕ' *to be the formula obtained by replacing* $\vee$*-nodes with* $\wedge$*-nodes and* $\wedge$*-nodes with* $\vee$*-nodes in* ϕ*. Then for all* $x \in \{0,1\}^N$*, if* $\bar{x}$ *denotes the bitwise complement of* x*, then* $\phi'(x) = \neg\phi(\bar{x})$*. Furthermore up to an isomorphism that maps* s *to* s'*,* t *to* t'*, and an edge labeled by any label* λ *to an edge labeled by* λ*, we have* $G'_\phi = G_{\phi'}$ *and* $G'_\phi(x) = G_{\phi'}(\bar{x})$*.*

Proof. The first part of the proof is by induction. Suppose ϕ has depth 0, so $\phi = x_i$ for some variable x_i. Then $\phi'(x) = \phi(x) = \neg(\phi(\bar{x}))$. So suppose $\phi = \phi_1 \wedge \cdots \wedge \phi_l$. Then $\phi' = \phi'_1 \vee \cdots \vee \phi'_l$. Then by the induction hypothesis,

$$\phi'(x) = \phi'_1(x) \vee \cdots \vee \phi'_l(x) = (\neg\phi_1(\bar{x})) \vee \cdots \vee (\neg\phi_l(\bar{x})) = \neg(\phi_1(\bar{x}) \wedge \cdots \wedge \phi_l(\bar{x})) = \neg\phi(\bar{x}) \tag{63}$$

where the second to last equality is de Morgan's law. The case $\phi = \phi_1 \vee \cdots \vee \phi_l$ is similar.

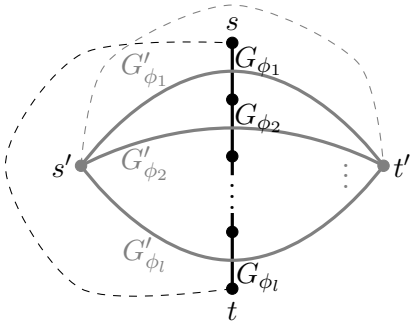


Figure 7: $\overline{G}_\phi$ shown in black, and its dual, $\overline{G}_{\phi'}$, shown in grey. The thick lines represent graphs. Edges in G_{ϕ_i} are dual to edges in G'_{ϕ_i}, and the dotted edge $(\{s,t\},\emptyset)$ is dual to $(\{s',t'\},\emptyset)$.

We will now prove that $\overline{G}^\dagger_\phi = \overline{G}_{\phi'}$, and furthermore, dual edges have the same label, by induction on the depth of ϕ, from which the result follows immediately.

If $\phi = x_i$ is a depth-0 formula, then $\phi' = x_i$. In that case, G_ϕ is just an edge from s to t, labeled by x_i, and G'_ϕ is just an edge from s' to t' labeled x_i, so $G'_\phi = G_{\phi'}$.

For the inductive step, to show that $\overline{G}_\phi$ and $\overline{G}_{\phi'}$ are dual, and therefore $G'_\phi = G_{\phi'}$. It suffices to exhibit a bijection $\zeta : V(\overline{G}_{\phi'}) \to F(\overline{G}_\phi)$ such that $(\{u,v\}, x_j) \in E(\overline{G}_{\phi'})$ if and only if the faces $\zeta(u)$ and $\zeta(v)$ are separated by an edge in $E(\overline{G}_\phi)$ with the label x_j. We first consider the case that $\phi = \phi_1 \wedge \cdots \wedge \phi_l$, so $\phi' = \phi'_1 \vee \cdots \vee \phi'_l$. Then, $\overline{G}_\phi$ consists of the graphs $G_{\phi_1}, \ldots, G_{\phi_l}$, chained together in series as in Figure 7, with an additional edge from s to t, so the faces of $\overline{G}_\phi$ are exactly all the interior faces of each G_{ϕ_i}, as well as the two faces on either side of the st-edge $(\{s,t\},\emptyset)$, which we will denote by $f^{s'}$ and $f^{t'}$. That is, adding an i to the label of each internal face of G_{ϕ_i}:

$$F(\overline{G}_\phi) = \{(i,f) : i \in [l], f \in F(\overline{G}_{\phi_i}) \setminus \{f^{s'}, f^{t'}\}\} \cup \{f^{s'}, f^{t'}\}, \tag{64}$$

since $F(\overline{G}_{\phi_i}) \setminus \{f^{s'}, f^{t'}\} = F(G_{\phi_i}) \setminus \{f^E\}$, where f^E is the external face. Since $\phi' = \phi'_1 \vee \cdots \vee \phi'_l$ we also have

$$V(\overline{G}_{\phi'}) = V(G_{\phi'}) = \{(i, v) : i \in [l], v \in V(G_{\phi'_i}) \setminus \{s, t\}\} \cup \{s', t'\}, \tag{65}$$

where we will use the labels s' and t' in anticipation of the isometry between G'_ϕ and $G_{\phi'}$.

By the induction hypothesis, for each $i \in [l]$, there exists a bijection $\zeta_i : V(\overline{G}_{\phi'_i}) \to F(\overline{G}_{\phi_i})$ such that for all $u, v \in V(\overline{G}_{\phi'_i}) = V(G_{\phi'_i})$, $(\{u, v\}, x_j) \in E(\overline{G}_{\phi'_i})$ if and only if $\zeta_i(u)$ and $\zeta_i(v)$ are faces separated by an edge with the label x_j. We define ζ by $\zeta(i, v) = (i, \zeta_i(v))$ for all $i \in [l]$ and $v \in V(G_{\phi'_i}) \setminus \{s, t\}$, $\zeta(s') = f^{s'}$, and $\zeta(t') = f^{t'}$. By the induction hypothesis, we can see that for any edge $(\{u, v\}, x_j) \in E(\overline{G}_{\phi'}) \setminus (\{s', t'\}, \emptyset)$, $\zeta(u)$ and $\zeta(v)$ are separated by an edge labeled x_j. This is because this edge is in one of the $G_{\phi'_i}$, and so it has a dual edge in G_{ϕ_i}, by the induction hypothesis (see Figure 7). The only other edge in $\overline{G}_{\phi'}$ is the edge $(\{s', t'\}, \emptyset)$, and $\zeta(s')$ and $\zeta(t')$ are exactly those faces on either side of $(\{s, t\}, \emptyset)$ in $\overline{G}_\phi$, completing the proof that $\overline{G}^\dagger_\phi = \overline{G}_{\phi'}$.

If $\phi = \phi_1 \vee \cdots \vee \phi_l$, then $\phi' = \phi'_1 \wedge \cdots \wedge \phi'_l$, and a nearly identical proof shows that $\overline{G}^\dagger_\phi = \overline{G}_{\phi'}$.

Now that we have shown an isomorphism between G'_ϕ and $G_{\phi'}$, note that $G'_\phi(x)$ is the subgraph of G'_ϕ that includes all those edges where $x_e = 0$. On the other hand $G_{\phi'}(x)$ is the graph that includes all those edges where $x_e = 1$. Taking the bitwise negation of x, we find that $G'_\phi(x) = G_{\phi'}(\bar{x})$. □

Claim 17. *If $\phi = \phi_1 \vee \phi_2 \vee \cdots \vee \phi_l$, then $G'_\phi(x)$ is formed by composing $\{G'_{\phi_i}(x)\}_i$ in series, and if $\phi = \phi_1 \wedge \phi_2 \wedge \cdots \wedge \phi_l$, then $G'_\phi(x)$ is formed by composing $\{G'_{\phi_i}(x)\}_i$ in parallel.*

Proof. If $\phi = \phi_1 \vee \cdots \vee \phi_l$, then $\phi' = \phi'_1 \wedge \cdots \wedge \phi'_l$. From Lemma 35, $G'_\phi(x) = G_{\phi'}(\bar{x})$, which using Definition 34 is composed of $\{G_{\phi'_i}(\bar{x})\}_{i=1}^l$ in series. But using the isomorphism of Lemma 35 again, this is just $\{G'_{\phi_i}(x)\}_{i=1}^l$ composed in series. The proof for $\phi = \phi_1 \wedge \cdots \wedge \phi_l$ is similar. □

$G_\phi(x)$ ɒ $G'_\phi(x)$

$\phi(x)$:

Lemma 14. *Let ϕ be any* AND-OR *formula on N variables. For every $x \in \{0, 1\}^N$, there exists a path from s to t in $G_\phi(x)$ if and only if $\phi(x) = 1$. Furthermore, for every $x \in \{0, 1\}^N$, there exists a path from s' to t' in $G'_\phi(x)$ if and only if $\phi(x) = 0$.*

Proof. We will prove the statement by induction on the depth of ϕ. If $\phi = x_j$ has depth 0, then G_ϕ is just an edge $(\{s, t\}, x_j)$, and G'_ϕ is just an edge $(\{s', t'\}, x_j)$. Thus s and t are connected in $G_\phi(x)$ if and only if $x_j = 1$, in which case ϕ evaluates to 1, and s' and t' are connected in $G_{\phi'}$ if and only if $x_j = 0$, in which case ϕ evaluates to 0.

If $\phi = \phi_1 \wedge \cdots \wedge \phi_l$, then G_ϕ consists of $G_{\phi_1}, \ldots, G_{\phi_l}$ connected in series from s to t, and moreover, $G_\phi(x)$ consists of $G_{\phi_1}(x), \ldots, G_{\phi_l}(x)$ connected in series from s to t. Thus an st-path in $G_\phi(x)$ consists of an st-path in $G_{\phi_1}(x)$, followed by an st-path in $G_{\phi_2}(x)$, etc., up to an st-path in $G_{\phi_l}(x)$. Thus, s and t are connected in $G_\phi(x)$ if and only if s and t are connected in each $G_{\phi_1}(x), \ldots, G_{\phi_l}(x)$, which happens if and only if $\phi_1(x) \wedge \cdots \wedge \phi_l(x) = 1$.

On the other hand, by Claim 17, G'_ϕ consists of $G'_{\phi_1}, \ldots, G'_{\phi_l}$ connected in parallel between s' and t'. So any $s't'$-path in $G'_\phi(x)$ is an $s't'$-path in one of the $G'_{\phi_i}(x)$, which is equivalent to an st-path in one of $G_{\phi'_i}(\bar{x})$. Thus, by Lemma 35 s' and t' are connected in

$G'_\phi(x)$ if and only if $\phi'_1(\bar{x}) \vee \cdots \vee \phi'_l(\bar{x}) = \neg\phi_1(x) \vee \cdots \vee \neg\phi_l(x) = 1$. By de Morgan's law is true if and only if $\phi(x) = \phi_1(x) \wedge \cdots \wedge \phi_l(x) = 0$.
The case when $\phi = \phi_1 \vee \cdots \vee \phi_l$ is similar. □

C Classical Lower Bound on Class of Promise Boolean Formulas

[illegible] AND-OR [illegible]

[illegible] f, [illegible] $R(f)$, [illegible] $R(f) = \Omega(RS(f))$, [illegible] $RS(f)$ [illegible]

[illegible] $f \circ g$, $RS(f \circ g) \geq (RS(f)RS(g))$.

[illegible] $f : D \to \{0,1\}$, [illegible] $D \subseteq \{0,1\}^N$, [illegible] $f_{\text{sab}} : D_{\text{sab}} \to \{0,1\}$, [illegible]

$$D_{\text{sab}} = \{x \in \{0,1,*\}^N \cup \{0,1,\dagger\}^N : x \text{ is consistent with } y, y' \in D, \text{ s.t. } f(y) \neq f(y')\}. \tag{66}$$

[illegible] $x \in \{0,1,*,\dagger\}^N$ [illegible] $y \in \{0,1\}^N$ [illegible] $x_i = y_i$ [illegible] $i \in [N]$ [illegible] $x_i \in \{0,1\}$. [illegible] $f_{\text{sab}}(x) = 1$ [illegible] $x \in \{0,1,*\}^N$, [illegible] $f_{\text{sab}}(x) = 0$ [illegible] $x \in \{0,1,\dagger\}^N$. [illegible] $RS(f) = R_0(f_{\text{sab}})$, [illegible] $R_0(f)$ [illegible] f. [illegible]

[illegible] $\text{AND}|_{D_{N,h}}$ [illegible] $\text{OR}|_{D'_{N,h}}$:

Lemma 36. *$RS\left(\text{OR}|_{D'_{N,h}}\right) = RS\left(\text{AND}|_{D_{N,h}}\right) = \Omega(N/h)$.*

Proof. For $x \in [D'_{N,h}]_{\text{sab}}$ to be consistent with $y, y' \in D'_{N,h}$ such that $\text{OR}(y) \neq \text{OR}(y')$, we must have that $x \in \{0,*\}^N \cup \{0,\dagger\}^N$. Furthermore, the number of $*$'s or $\dagger$'s in x must be at least h. Thus the sabotaged problem reduces to finding at least one marked item out of n, promised there are at least h marked items. The randomized bounded-error query complexity of this task is $\Omega(N/h)$, and so by Theorem 3 in [5],

$$RS\left(\text{OR}|_{D'_{N,h}}\right) = R_0\left((\text{OR}|_{D'_{N,h}})_{\text{sab}}\right) = \Omega\left(R\left((\text{OR}|_{D'_{N,h}})_{\text{sab}}\right)\right) = \Omega(N/h). \tag{67}$$

The proof for AND is similar. □

[illegible]

Corollary 37. *Let $\phi = \phi_1 \circ \phi_2 \circ \cdots \circ \phi_l$, where for each $i \in [l]$, $\phi_i = \text{OR}|_{D'_{N_i,h_i}}$ or $\phi_i = \text{AND}|_{D_{N_i,h_i}}$. Then $R(\phi) = \Omega\left(\prod_{i=1}^l N_i/h_i\right)$.*

[illegible] AND-OR [illegible]

Lemma 38. *Let $\phi = \phi_1 \circ \phi_2 \circ \cdots \circ \phi_l$, where for each $i \in [l]$, $\phi_i = \text{OR}|_{D'_{N_i,h_i}}$ or $\phi_i = \text{AND}|_{D_{N_i,h_i}}$. Let D be the domain of ϕ. Then*

$$\frac{\prod_{i=1}^l N_i}{\prod_{i=1}^l h_i} = \left(\max_{x \in D:\phi(x)=1} R_{s,t}(G_\phi(x))\right)\left(\max_{x \in D:\phi(x)=0} R_{s,t}(G'_\phi(x))\right). \tag{68}$$

Proof. The proof follows by induction on the number of compositions. First suppose that $\phi = \text{OR}|_{D'_{N,h}}$. Then G_ϕ consists of N edges connected in parallel between s and t, and G'_ϕ consists of N edges connected in series. The only input x such that $\phi(x) = 0$ is the all zeros input. Therefore $\max_{x \in D:\phi(x)=0} R_{s,t}(G'_\phi(x)) = N$. Now notice (using Claim 5) that $R_{s,t}(G_\phi(x)) = 1/|x|$. However because of the domain of OR_{N_i,h_i}, inputs x have $|x| \geq h$, so $\max_{x \in D:\phi(x)=1} R_{s,t}(G_\phi(x)) = 1/h$. Thus

$$N/h = \left(\max_{x \in D:\phi(x)=1} R_{s,t}(G_\phi(x)) \right) \left(\max_{x \in D:\phi(x)=0} R_{s,t}(G'_\phi(x)) \right). \tag{69}$$

A similar analysis holds for the base case $\phi = \text{AND}|_{D_{N,h}}$.

Now for the inductive step, let $\phi = \phi_1 \circ \xi$ for $\xi = \phi_2 \circ \cdots \circ \phi_l$, where for each i, ϕ_i is either $\text{OR}|_{D'_{N_i,h_i}}$ or $\text{AND}|_{D_{N_i,h_i}}$. Let D_ξ be the domain of ξ and let $x^j \in D_\xi$ denote the bits of x that are input to the j^{th} copy of ξ. Suppose first that $\phi_1 = \text{OR}|_{D'_{N_1,h_1}}$. G'_ϕ is formed by taking the N_1 graphs G'_ξ and connecting them in series. The only way $\phi(x) = 0$ is if the input $x^j \in D_\xi$ to each of the ξ functions satisfies $\xi(x^j) = 0$, so by Claim 5

$$\max_{x \in D:\phi(x)=0} R_{s,t}(G'_\phi(x)) = N_1 \max_{y \in D_\xi:\xi(y)=0} R_{s,t}(G'_\xi(y)). \tag{70}$$

On the other hand, G_ϕ is formed by taking N_1 graphs G_ξ and connecting them in parallel. Using Claim 5, if $x^j \in D_\xi$ is the input to j^{th} function ξ, we have

$$R_{s,t}(G_\phi(x)) = \left(\sum_{j=1}^{N} \frac{1}{R_{s,t}(G_\xi(x^j))} \right)^{-1}. \tag{71}$$

Thus larger values for $R_{s,t}(G_\phi(x))$ come from cases where $R_{s,t}(G_\xi(x^j))$ are large. Now

$$\max_{x \in D_\xi} R_{s,t}(G_\xi(x)) = \infty, \tag{72}$$

which occurs when $\xi(y) = 0$. Because of the promise on the domain of ϕ_1, there must be at least h_1 of the N_1 subformulas ξ that evaluate to 1. On each of those subformulas, we want to have an input $x^j \in D_\xi$ that maximizes the effective resistance of that subformula. Therefore, we have

$$\max_{x \in D:\phi(x)=1} R_{s,t}(G_\phi(x)) = \left(\frac{h_1}{\max_{y \in D_\xi:\xi(y)=1} R_{s,t}(G_{\xi_j}(y)} \right)^{-1} = \frac{\max_{y \in D_\xi:\xi(y)=1} R_{s,t}(G_{\xi_j}(y))}{h_1}. \tag{73}$$

Therefore, using the inductive assumption,

$$\begin{aligned} &\left(\max_{x \in D:\phi(x)=1} R_{s,t}(G_\phi(x)) \right) \left(\max_{x \in D:\phi(x)=0} R_{s,t}(G'_\phi(x)) \right) \\ =&\frac{N_1}{h_1} \max_{y \in D_\xi:\xi(y)=1} R_{s,t}(G_{\xi_j}(y^j)) \max_{y \in D_\xi:\xi(y)=0} R_{s,t}(G'_\xi(y)) = \frac{\prod_{i=1}^{l} N_i}{\prod_{i=1}^{l} h_i}. \end{aligned} \tag{74}$$

The inductive step for $\phi_1 = \text{AND}|_{D_{N,h}}$ is similar. □

D NAND-tree Proofs

D.1 Relationship Between Faults and Effective Resistance

Lemma 22. *For any $x \in \{0,1\}^{2^d}$, if d is even, then we have $R_{s,t}(G_{\mathrm{NAND}_d}(x)) \leq \mathcal{F}_A(x)$ and $R_{s',t'}(G'_{\mathrm{NAND}_d}(x)) \leq \mathcal{F}_B(x)$, while if d is odd, we have $R_{s,t}(G_{\mathrm{NAND}_d}(x)) \leq 2\mathcal{F}_A(x)$ and $R_{s',t'}(G'_{\mathrm{NAND}_d}(x)) \leq 2\mathcal{F}_B(x)$.*

Proof. We will give a proof for $\mathcal{F}_A(x)$; the case of $\mathcal{F}_B(x)$ is similar.

First, $R_{s,t}(G_{\mathrm{NAND}_d}(x)) = \infty$ if and only if s and t are not connected in $G_{\mathrm{NAND}_d}(x)$, which, by Lemma 14, occurs if and only if x is a 0-instance. This means exactly that x is not A-winnable, which, by Eq. (28), holds if and only if $\mathcal{F}_A(x) = \infty$. Thus, suppose this is not the case, so $\mathcal{F}_A(x) < \infty$.

The rest of the proof is by induction. We need to look at both odd and even cases. For the case of $d = 0$, the only A-winnable input in $\{0,1\}^{2^0}$ is $x = 1$. In that case, using Eq. (28), $\mathcal{F}_A(x) = 1$, since there are no decision nodes for Player A, and since $G_{\mathrm{NAND}_0}(x)$ is just a single edge from s to t, $R_{s,t}(G_{\mathrm{NAND}_0}(x)) = 1$.

Let $x \in \{0,1\}^{2^d}$ be any A-winnable input with $d > 1$. We let x^0 be the first 2^{d-1} bits of x and x^1 be the last 2^{d-1} bits of x, so $x = (x^0, x^1)$.

We first consider odd $d > 1$. Using the definition of G_ϕ from Section 4, and the fact that for d odd, the root node is an $\wedge$-node, we see that $G_{\mathrm{NAND}_d}(x)$ consists of $G_{\mathrm{NAND}_{d-1}}(x^0)$ and $G_{\mathrm{NAND}_{d-1}}(x^1)$ connected in series, so by Claim 5

$$R_{s,t}(G_{\mathrm{NAND}_d}(x)) = R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^0)) + R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^1)). \tag{75}$$

Now the root can not be a fault, because it is a decision node for Player B, but we know the tree is A-winnable, so no choice Player B makes would allow her to win the game. Therefore, both subtrees connected to the root node must be A-winnable. Using Eq. (28) we have

$$\mathcal{F}_A(x^0) + \mathcal{F}_A(x^1) \leq \max_{b\in\{0,1\}} 2\mathcal{F}_A(x^b) = 2\mathcal{F}_A(x). \tag{76}$$

Combining Eqs. (75) and (76) and the inductive assumption for even depth trees, we have for odd d,

$$R_{s,t}(G_{\mathrm{NAND}_d}(x)) \leq 2\mathcal{F}_A(x). \tag{77}$$

Now we consider the case that d is even, so the root is a decision node for Player A. Consequently, the root node is a $\vee$-node, so by Claim 5

$$R_{s,t}(G_{\mathrm{NAND}_d}(x)) = \left(\frac{1}{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^0))} + \frac{1}{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^1))}\right)^{-1}. \tag{78}$$

Suppose the root is a fault. Without loss of generality, let's assume the subtree with input x^0 is not A-winnable. Then $R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^0)) = \infty$ so Eq. (78) becomes

$$R_{s,t}(G_{\mathrm{NAND}_d}(x)) = R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^1)). \tag{79}$$

Using the inductive assumption for odd depth trees, Eq. (28), and the fact that the root is a fault, we have

$$R_{s,t}(G_{\mathrm{NAND}_d}(x)) \leq 2\mathcal{F}(x^1) = \mathcal{F}(x). \tag{80}$$

If the root is not a fault, then both $R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^0))$ and $R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^1))$ are finite, so from (78), and using the inductive assumption, we have

$$\begin{aligned} R_{s,t}(G_{\mathrm{NAND}_d}(x)) &\leq \frac{1}{2}\max\{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^0)), R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^1))\} \\ &\leq \max\{\mathcal{F}(x^0), \mathcal{F}(x^1)\} = \mathcal{F}(x). \end{aligned} \tag{81}$$

A similar analysis for $\mathcal{F}_B(x)$ completes the proof. □

D.2 Estimating Effective Resistances

[illegible] $\mathcal{O}_x$. [illegible] $G_\phi(x)$ [illegible] $G'_\phi(x)$.

[illegible] *approximate positive and negative witness sizes,* $\tilde{w}_+(x,P)$ [illegible] $\tilde{w}_-(x,P)$. [illegible] $|w\rangle \in H(x)$ [illegible] $\omega A \Pi_{H(x)} = 0$ [illegible]

Definition 39 (Approximate Positive Witness)**.** *For any span program* P *on* $\{0,1\}^N$ *and* $x \in \{0,1\}^N$*, we define the* positive error of x in P *as:*

$$e_+(x) = e_+(x,P) := \min\left\{\left\|\Pi_{H(x)^\perp}|w\rangle\right\|^2 : A|w\rangle = \tau\right\}. \tag{82}$$

We say $|w\rangle$ *is an* approximate positive witness *for* x *in* P *if* $\left\|\Pi_{H(x)^\perp}|w\rangle\right\|^2 = e_+(x)$ *and* $A|w\rangle = \tau$*. We define the* approximate positive witness size *as*

$$\tilde{w}_+(x) = \tilde{w}_+(x,P) := \min\left\{\||w\rangle\|^2 : A|w\rangle = \tau, \left\|\Pi_{H(x)^\perp}|w\rangle\right\|^2 = e_+(x)\right\}. \tag{83}$$

[illegible] $x \in P_1$, [illegible] $e_+(x) = 0$. [illegible] x [illegible] $\tilde{w}_+(x) = w_+(x)$. [illegible] 0

Definition 40 (Approximate Negative Witness)**.** *For any span program* P *on* $\{0,1\}^N$ *and* $x \in \{0,1\}^N$*, we define the* negative error of x in P *as:*

$$e_-(x) = e_-(x,P) := \min\left\{\left\|\omega A \Pi_{H(x)}\right\|^2 : \omega \in \mathcal{L}(U,\mathbb{R}), \omega\tau = 1\right\}. \tag{84}$$

Any ω *such that* $\left\|\omega A \Pi_{H(x)}\right\|^2 = e_-(x,P)$ *is called an* approximate negative witness *for* x *in* P*. We define the* approximate negative witness size *as*

$$\tilde{w}_-(x) = \tilde{w}_-(x,P) := \min\left\{\|\omega A\|^2 : \omega \in \mathcal{L}(U,\mathbb{R}), \omega\tau = 1, \left\|\omega A \Pi_{H(x)}\right\|^2 = e_-(x,P)\right\}. \tag{85}$$

[illegible] $x \in P_0$, [illegible] $e_-(x) = 0$. [illegible] x [illegible] $\tilde{w}_-(x) = w_-(x)$. [illegible] 0

Theorem 41 ([17]). *Fix $X \subseteq \{0,1\}^N$ and $f : X \to \mathbb{R}_{\geq 0}$. Let P be a span program such that for all $x \in X$, $f(x) = w_+(x, P)$ and define $\widetilde{W}_- = \widetilde{W}_-(P, f) = \max_{x \in X} \tilde{w}_-(x, P)$. There exists a quantum algorithm that estimates f to relative error ε and that uses $\widetilde{O}\left(\frac{1}{\varepsilon^{3/2}}\sqrt{w_+(x)\widetilde{W}_-}\right)$ queries. Similarly, let P be a span program such that for all $x \in X$, $f(x) = w_-(x, P)$ and define $\widetilde{W}_+ = \widetilde{W}_+(P, f) = \max_{x \in X} \tilde{w}_+(x, P)$. Then there exists a quantum algorithm that estimates f to accuracy ε and that uses $\widetilde{O}\left(\frac{1}{\varepsilon^{3/2}}\sqrt{w_-(x)\widetilde{W}_+}\right)$ queries.*

[illegible] $P_{G,c}$ [illegible] (6), [illegible] $G = G_\phi$. [illegible] c [illegible] G. [illegible] $P_{G,c}$ [illegible] P_G. [illegible] $\widetilde{W}_+(P_{G_\phi})$ [illegible] $\widetilde{W}_-(P_{G_\phi})$. [illegible]

Lemma 42. *For any formula ϕ, its $\wedge$-depth is the largest number of $\wedge$-labeled nodes on any path from the root to a leaf. Let ϕ be any* AND-OR *formula with maximum fan-in l, $\wedge$-depth $d_\wedge$, and $\vee$-depth $d_\vee$. Then $\widetilde{W}_+(P_{G_\phi}) \leq \frac{1}{2}l^{d_\wedge}$ and $\widetilde{W}_-(P_{G_\phi}) \leq 2l^{d_\vee}$.*

[illegible]

Lemma 25 (`Est` Algorithm). *Let ϕ be an* AND-OR *formula with constant fan-in l, $\vee$-depth $d_\vee$ and $\wedge$-depth $d_\wedge$. Then the quantum query complexity of estimating $R_{s,t}(G_\phi(x))$ (resp. $R_{s,t}(G'_\phi(x))$) to relative accuracy ϵ is $\widetilde{O}\left(\frac{1}{\varepsilon^{3/2}}\sqrt{R_{s,t}(G_\phi(x))l^{d_\vee}}\right)$ (resp. $\widetilde{O}\left(\frac{1}{\varepsilon^{3/2}}\sqrt{R_{s,t}(G'_\phi(x))l^{d_\wedge}}\right)$).*

Proof of Lemma 25. By Theorem 41, since $R_{s,t}(G_\phi(x)) = \frac{1}{2}w_+(x, P_{G_\phi})$ (Lemma 11), we can estimate this quantity using a number of queries that depends on $\widetilde{W}_-(P_{G_\phi})$. By Lemma 42, we have that $\widetilde{W}_-(P_{G_\phi}) \leq 2l^{d_\vee}$, so we can estimate $w_+(x) = R_{s,t}(G_\phi(x))$ in $\widetilde{O}\left(\frac{1}{\varepsilon^{2/3}}\sqrt{w_+(x)\widetilde{W}_-^{1/2}}\right) = \widetilde{O}\left(\frac{1}{\varepsilon^{2/3}}\sqrt{R_{s,t}(G_\phi(x))l^{d_\vee}}\right)$ queries. Similarly, $R_{s,t}(G'_\phi(x)) = 2w_-(x, P_{G_\phi})$ for all 0-instances, and $\widetilde{W}_+ \leq \frac{1}{2}l^{d_\wedge}$, so we can estimate $R_{s,t}(G'_\phi(x))$ in $\widetilde{O}\left(\frac{1}{\varepsilon^{2/3}}\sqrt{R_{s,t}(G'_\phi(x))l^{d_\wedge}}\right)$ queries. □

[illegible] *st* [illegible] G_ϕ, [illegible] $\wedge$ [illegible] ϕ. [illegible]

Claim 43. *Let ϕ be an* AND-OR *formula with constant fan-in l. If ϕ has $\wedge$-depth $d_\wedge$, then the longest self-avoiding path connecting s and t in G_ϕ has length at most $l^{d_\wedge}$.*

Proof. We will prove the statement by induction. If ϕ has $\wedge$-depth $d_\wedge = 0$, then it has no $\wedge$-nodes. Thus, it is easy to see that G_ϕ has only two vertices, s and t, with some number of edges connecting them, so every st-path has length 1.

Suppose ϕ has $\wedge$-depth $d_\wedge > 0$. First, suppose $\phi = \phi_1 \wedge \cdots \wedge \phi_l$. Then since G_ϕ consists of $G_{\phi_1}, \ldots, G_{\phi_l}$ connected in series, any st-path in G_ϕ consists of an st-path in G_{ϕ_1}, followed by an st-path in G_{ϕ_2}, etc. up to an st-path in G_{ϕ_l}, so if $d_\wedge(\phi_i)$ is the $\wedge$-depth of ϕ_i, then the longest st-path in G_ϕ has length at most:

$$l^{d_\wedge(\phi_1)} + \cdots + l^{d_\wedge(\phi_l)} \leq ll^{d_\wedge - 1} = l^{d_\wedge}. \tag{86}$$

If $\phi = \phi_1 \vee \cdots \vee \phi_l$, then $\max_i d_\wedge(\phi_i) = d_\wedge(\phi) = d_\wedge$, and G_ϕ consists of $G_{\phi_1}, \ldots, G_{\phi_l}$, connected in parallel. Any self-avoiding st-path must include exactly one edge adjacent to s and one edge adjacent to t. However, any path that includes an edge from G_{ϕ_i} and G_{ϕ_j} for $i \neq j$ must go through s or t, so it must have more than one edge adjacent to s, or more than one edge adjacent to t, so such a path can never be a self-avoiding st-path. Thus, any self-avoiding st-path must be contained completely in one of the G_{ϕ_i}. The longest such path is thus the longest self-avoiding st-path in any of the G_{ϕ_i}, which, by induction, is $\max_i l^{d_\wedge(\phi_i)} = l^{d_\wedge}$. □

Proof of Lemma 42. To begin, we will prove the upper bound on $\widetilde{W}_+$. Suppose $|\tilde{w}\rangle$ is an optimal approximate positive witness for x. By Claim 30, if $|\tilde{w}\rangle$ is an approximate positive witness, then since $A|\tilde{w}\rangle = \tau$, and c has unit value on all edges of G, $\theta(u,v,\lambda) = \langle u,v,\lambda|\tilde{w}\rangle - \langle v,u,\lambda|\tilde{w}\rangle$ is a unit flow on G. Since $|\tilde{w}\rangle$ is an approximate positive witness for x, it has minimal error for x, so it minimizes $\left\|\Pi_{H(x)^\perp}|\tilde{w}\rangle\right\|^2$, and since it is optimal, it minimizes $\||\tilde{w}\rangle\|^2$ over all approximate positive witnesses. Define $|\theta\rangle = \sum_{(u,v,\lambda)\in\vec{E}(G)} \theta(u,v,\lambda)|u,v,\lambda\rangle$, so we know that $\frac{1}{2}|\theta\rangle$ also maps to τ under A, so is also a positive witness in P_{G_ϕ}.Then we have

$$\left\|\Pi_{H(x)^\perp}|\theta\rangle\right\|^2 = 2 \sum_{\substack{(u,v,\lambda)\in \\ \vec{E}(G)\setminus\vec{E}(G(x))}} \langle u,v,\lambda|\tilde{w}\rangle^2 - 2 \sum_{\substack{(u,v,\lambda)\in \\ \vec{E}(G)\setminus\vec{E}(G(x))}} \langle u,v,\lambda|\tilde{w}\rangle\langle v,u,\lambda|\tilde{w}\rangle \leq \left\|2\Pi_{H(x)^\perp}|\tilde{w}\rangle\right\|^2, \tag{87}$$

where the last inequality uses Cauchy-Schwarz, so $\frac{1}{2}|\theta\rangle$ is also an approximate positive witness for x. Similarly,

$$\||\theta\rangle\|^2 \leq \|2|\tilde{w}\rangle\|^2, \tag{88}$$

so $\frac{1}{2}|\theta\rangle$ is optimal.

By Claim 29, we can consider a decomposition of $|\theta\rangle$ into self-avoiding paths p_i and cycles c_i such that all cycles are disjoint from all paths, $|\theta\rangle = \sum_{i=1}^r \alpha_i|p_i\rangle + \sum_{i=1}^{r'} \beta_i|c_i\rangle$, where for each i,

$$|p_i\rangle = \sum_{j=1}^{L_i} |u_j^{(i)}, u_{j+1}^{(i)}, \lambda_{i,j}\rangle - \sum_{j=1}^{L_i} |u_{j+1}^{(i)}, u_j^{(i)}, \lambda_{i,j}\rangle, \tag{89}$$

$$|c_i\rangle = \sum_{j=1}^{L_i'} |v_j^{(i)}, v_{j+1}^{(i)}, \lambda'_{i,j}\rangle - \sum_{j=1}^{L_i'} |v_{j+1}^{(i)}, v_j^{(i)}, \lambda'_{i,j}\rangle \tag{90}$$

where $v_{L_j'+1}^{(i)} = v_1^{(i)}$ and $\{\lambda_{i,j}\}_{i,j} \cap \{\lambda'_{i,j}\}_{i,j} = \emptyset$. It's easy to see (in the case of unit edge weights) that $A|c_i\rangle = 0$ for all i, so

$$A\frac{1}{2}\sum_{i=1}^r \alpha_i|p_i\rangle = A\frac{1}{2}|\theta\rangle = \tau. \tag{91}$$

Let $|\theta'\rangle = \sum_{i=1}^r \alpha_i|p_i\rangle$. Then since c_i and p_j have no common edges, we have $\langle c_i|p_j\rangle = 0$, and also $\langle c_i|(I - \Pi_{H(x)})|p_j\rangle = 0$, so the error of $\frac{1}{2}|\theta'\rangle$ is $\frac{1}{4}\left\|\Pi_{H(x)^\perp}|\theta'\rangle\right\|^2 \leq \frac{1}{4}\left\|\Pi_{H(x)^\perp}|\theta\rangle\right\|^2$, so $\frac{1}{2}|\theta'\rangle$ also has minimal error. Furthermore, $\||\theta'\rangle\|^2 \leq \||\theta\rangle\|^2$, with equality if and only if there are no cycles in the decomposition. By the optimality of $\frac{1}{2}|\theta\rangle$ as an approximate

positive witness for x, we can conclude that $|\theta\rangle = \sum_{i=1}^{r} \alpha_i |p_i\rangle$, and since $A|p_i\rangle = 2\tau$ for all i, and $A|\theta\rangle = 2\tau$, we have $\sum_{i=1}^{r} \alpha_i = 1$. Then

$$\||\theta\rangle\|^2 \leq \max_i \||p_i\rangle\|^2 = \max_i 2L_i. \tag{92}$$

Since the longest self-avoiding st-path in G_ϕ has length at most $l^{d_\wedge}$, and each L_i is the length of a self-avoiding path in G_ϕ, we have $\tilde{w}_+(x, P_{G_\phi}) \leq \frac{1}{4} 2 l^{d_\wedge} = \frac{1}{2} l^{d_\wedge}$. Thus $\widetilde{W}_+(P_{G_\phi}) \leq \frac{1}{2} l^{d_\wedge}$.

Next we prove the bound on $\widetilde{W}_-$. A min-error approximate negative witness for x in P_{G_ϕ} is a function $\omega : V(G_\phi) \to \mathbb{R}$ such that $\omega\tau = \omega(s) - \omega(t) = 1$, and $\left\|\omega A \Pi_{H(x)}\right\|^2 = \sum_{(u,v,\lambda)\in\overrightarrow{E}(G_\phi(x))} (\omega(u) - \omega(v))^2$ is minimized. By Claim 31, since $\omega\tau = 1$, the function $\theta : \overrightarrow{E}(G'_\phi) \to \mathbb{R}$ defined by $\theta((u, v, \lambda)^\dagger) = \omega(u) - \omega(v)$ is a unit $s't'$-flow on $G'_\phi = G_{\phi'}$, and the witness complexity is

$$\|\omega A\|^2 = \sum_{(u,v,\lambda)\in\overrightarrow{E}(G_\phi)} (\omega(u) - \omega(v))^2 = \sum_{(u',v',\lambda)\in\overrightarrow{E}(G'_\phi)} \theta(u', v', \lambda)^2 = \||\theta\rangle\|^2 \tag{93}$$

where we create $|\theta\rangle$ from θ in the usual way. By an argument similar to the previous argument, if ω is an optimal approximate negative witness for x, then $\||\theta\rangle\|^2$ is upper bounded by twice the length of the longest self-avoiding $s't'$-path in $G'_\phi = G_{\phi'}$. By Lemma 35 and Claim 43, this is upper bounded by $2l^{d_\wedge(\phi')} = 2l^{d_\vee(\phi)}$, where $d_\wedge(\phi')$ is the $\wedge$-depth of ϕ', and $d_\vee = d_\vee(\phi)$ is the $\vee$-depth of ϕ. Thus $\tilde{w}_-(x, P_{G_\phi}) \leq 2l^{d_\vee}$, and so $\widetilde{W}_- \leq 2l^{d_\vee}$. □

D.3 Winning the NAND-tree

[illegible] NAND[illegible]

Lemma 26. *Let $x^0, x^1 \in \{0,1\}^{2^d}$ be instances of NAND_d with at least one of them a 1-instance. Let $N = 2^d$, and $w_{\min} = \min\{R_{s,t}(G_{\mathrm{NAND}_d}(x^0)), R_{s,t}(G_{\mathrm{NAND}_d}(x^1))\}$. Then $\mathtt{Select}(x^0, x^1)$ terminates after $\widetilde{O}\left(N^{1/4}\sqrt{w_{\min}}\right)$ queries to (x^0, x^1) and outputs b such that $R_{s,t}(G_{\mathrm{NAND}_d}(x^b)) \leq 2R_{s,t}(G_{\mathrm{NAND}_d}(x^{\bar{b}}))$ with bounded error.*

Proof. Since at least one of x^0 and x^1 is a 1-instance, using the description of `Select` in Section 5.2, at least one of the programs will terminate. Suppose without loss of generality that $\mathtt{Est}(x^0)$ is the first to terminate, outputting w_0. Then there are two possibilities: $\mathtt{Est}(x^1)$ does not terminate after $p(d)\sqrt{w_0}N^{1/4}$ steps, in which case, $R_{s,t}(G_{\mathrm{NAND}_d}(x^0)) \leq 2R_{s,t}(G_{\mathrm{NAND}_d}(x^1))$, and `Select` outputs 0; or $\mathtt{Est}(x^1)$ outputs w_1 before $p(d)\sqrt{w_0}N^{1/4}$ steps have passed and `Select` outputs b such that $w_b \leq w_{\bar{b}}$.

We will prove the first case by contradiction. Suppose

$$2R_{s,t}(G_{\mathrm{NAND}_d}(x^1)) < R_{s,t}(G_{\mathrm{NAND}_d}(x^0)). \tag{94}$$

Then $\mathtt{Est}(x^1)$ must terminate after

$$p(d)\sqrt{R_{s,t}(G_{\mathrm{NAND}_d}(x^1))}N^{1/4} \leq \frac{1}{\sqrt{2}} p(d)\sqrt{R_{s,t}(G_{\mathrm{NAND}_d}(x^0))}N^{1/4} \tag{95}$$

steps. In `Select`, we run `Est` to relative accuracy $\varepsilon = 1/3$, so we have

$$|w_0 - R_{s,t}(G_{\mathrm{NAND}_d}(x^0))| \leq \frac{1}{3} R_{s,t}(G_{\mathrm{NAND}_d}(x^0)), \tag{96}$$

and so

$$w_0 \geq \frac{2}{3} R_{s,t}(G_{\mathrm{NAND}_d}(x^0)). \tag{97}$$

Plugging Eq. (97) into Eq. (95), we have $\texttt{Est}(x^1)$ must terminate after $\frac{1}{\sqrt{2}} p(d) \sqrt{\frac{3}{2} w_0} N^{1/4} < p(d)\sqrt{w_0} N^{1/4}$ steps, which is a contradiction.

Thus, $R_{s,t}(G_{\mathrm{NAND}_d}(x^0)) \leq 2R_{s,t}(G_{\mathrm{NAND}_d}(x^1))$, so outputting 0 is correct. Furthermore, since we terminate after $p(d)\sqrt{w_0}N^{1/4} = \widetilde{O}(\sqrt{R_{s,t}(G_{\mathrm{NAND}_d}(x^0))}N^{1/4})$ steps, and since $R_{s,t}(G_{\mathrm{NAND}_d}(x^0)) = O(R_{s,t}(G_{\mathrm{NAND}_d}(x^1)))$, the running time is at most $\widetilde{O}\left(N^{1/4}\sqrt{w_{\min}}\right)$.

We now consider the second case, in which both programs output estimates w_0 and w_1, such that $|w_b - R_{s,t}(G_{\mathrm{NAND}_d}(x^b))| \leq \varepsilon R_{s,t}(G_{\mathrm{NAND}_d}(x^b))$ for $b = 0, 1$. Suppose $w_b \leq w_{\bar{b}}$. We then have

$$\frac{R_{s,t}(G_{\mathrm{NAND}_d}(x^b))}{R_{s,t}(G_{\mathrm{NAND}_d}(x^{\bar{b}}))} \leq \frac{R_{s,t}(G_{\mathrm{NAND}_d}(x^b))}{w_b} \frac{w_{\bar{b}}}{R_{s,t}(G_{\mathrm{NAND}_d}(x^{\bar{b}}))} \leq \frac{1+\varepsilon}{1-\varepsilon} = \frac{4/3}{2/3} = 2. \tag{98}$$

Thus $R_{s,t}(G_{\mathrm{NAND}_d}(x^b)) \leq 2R_{s,t}(G_{\mathrm{NAND}_d}(x^{\bar{b}}))$, as required. Furthermore, the running time of the algorithm is bounded by the running time of $\texttt{Est}(x^1)$, the second to terminate. We know that $\texttt{Est}(x^1)$ has running time at most $\widetilde{O}\left(\sqrt{R_{s,t}(G_{\mathrm{NAND}_d}(x^1))}N^{1/4}\right)$ steps, and by assumption, $\texttt{Est}(x^1)$ terminated after less than $p(d)\sqrt{w_0}N^{1/4} = \widetilde{O}\left(\sqrt{R_{s,t}(G_{\mathrm{NAND}_d}(x^0))}N^{1/4}\right)$ steps, so the total running time is at most $\widetilde{O}\left(N^{1/4}\sqrt{w_{\min}}\right)$. □

Theorem 27. *Let $x \in \{0,1\}^N$ for $N = 2^d$ be an A-winnable input to NAND_d. At every node v where Player A makes a decision, let Player A use the* `Select` *algorithm in the following way. Let v_0 and v_1 be the two children of v, with inputs to the respective subtrees of v_0 and v_1 given by x^0 and x^1 respectively. Then Player A moves to v_b where b is the outcome that occurs a majority of times when $\texttt{Select}(x^0, x^1)$ is run $O(\log d)$ times. Then if Player B, at his decision nodes, chooses left and right with equal probability, Player A will win the game with probability at least $2/3$, and will use $\widetilde{O}\left(N^{1/4}\sqrt{R_{s,t}(G_{\mathrm{NAND}_d}(x))}\right)$ queries on average, where the average is taken over the randomness of Player B's choices.*

Proof. First note that Player A must make $O(d)$ choices over the course of the game. We amplify Player A's probability of success by repeating `Select` at each decision node $O(\log d)$ times and taking the majority. Then the probability that Player A chooses the wrong direction at any node is $O(1/d)$, and we ensure that her probability of choosing the wrong direction over the course of the algorithm is $< 1/3$. From here on, we analyze the error free case.

Let $p(d)$ be a non-decreasing polynomial function in d such that `Select`, on inputs $x^0, x^1 \in \{0,1\}^{2^d}$, terminates in at most $p(d)2^{d/4}\sqrt{\min\{R_{s,t}(G_{\mathrm{NAND}_d}(x^0)), R_{s,t}(G_{\mathrm{NAND}_d}(x^1))\}}$ queries. Then we will prove that for trees of odd depth d, the expected number of queries by Player A over the course of the game is at most $p(d)2^{d/4+5}\sqrt{R_{s,t}(G_{\mathrm{NAND}_d}(x))}$, while for even depth trees, it is at most $p(d)2^{d/4+11/2}\sqrt{R_{s,t}(G_{\mathrm{NAND}_d}(x))}$, thus proving the main result.

We prove the result by induction on the depth of the tree. For depth zero trees, there are no decisions, $N = R_{s,t}G_{\mathrm{NAND}_0}(x) = 1$, so the result holds.

For the inductive case, we treat odd and even depth cases separately. First consider an instance of NAND_d with $d > 0$, d odd. Thus $\mathrm{NAND}_d(x) = \mathrm{NAND}_{d-1}(x^0) \wedge \mathrm{NAND}_{d-1}(x^1)$,

where $x = (x^0, x^1)$. Because the root is at odd distance from the leaves, the root is a decision node for Player B. Because we are in an A-winnable tree, no matter which choice Player B makes, we will end up at an A-winnable subtree of depth $d-1$, so the inductive assumption holds for those trees. That is, the expected number of queries for Player A must make to win the subtree with input x^b (for $b \in \{0,1\}$) averaged over Player B's choices is at most

$$p(d-1)2^{(d-1)/4+11/2}\sqrt{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^b))}. \tag{99}$$

We are assuming that Player B chooses left and right with equal probability. Thus, the expected number of queries that Player A must make over Player B's choices throughout the game is at most

$$\begin{aligned}
&\frac{1}{2}\Big(p(d-1)2^{(d-1)/4+11/2}\sqrt{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^0))} \\
&\qquad +p(d-1)2^{(d-1)/4+11/2}\sqrt{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^1))}\Big) \\
&\le p(d-1)2^{(d-1)/4+11/2}\sqrt{\frac{1}{2}\left(R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^0)) + R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^1))\right)} \quad \text{by Jensen's ineq.,} \\
&= p(d-1)2^{(d-1)/4+11/2}\sqrt{\frac{1}{2}R_{s,t}(G_{\mathrm{NAND}_d}(x))} \quad \text{by Claim 5,} \\
&\le p\left(d\right)2^{d/4-1/4+11/2-1/2}\sqrt{R_{s,t}(G_{\mathrm{NAND}_d}(x))} \\
&\le p\left(d\right)2^{d/4+5}\sqrt{R_{s,t}(G_{\mathrm{NAND}_d}(x))},
\end{aligned} \tag{100}$$

proving the case for odd d.

Now consider an instance of NAND_d with $d > 0$, d even. Thus $\mathrm{NAND}_d(x) = \mathrm{NAND}_{d-1}(x^0) \vee \mathrm{NAND}_{d-1}(x^1)$, where $x = (x^0, x^1)$. Because the root is at even distance from the leaves, the root is a decision node for Player A. Player A runs $\texttt{Select}(x^0, x^1)$, which returns $b \in \{0,1\}$ such that (by Lemma 26)

$$R_{s,t}(G_{\mathrm{NAND}^{d-1}}(x^b)) \le 2R_{s,t}(G_{\mathrm{NAND}^{d-1}}(x^{\bar{b}})), \tag{101}$$

which requires at most

$$\min_{b^* \in \{0,1\}} p\left(d-1\right)2^{(d-1)/4}\sqrt{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^{b^*}))} \tag{102}$$

queries.

After making the choice to move to the subtree with input x^b, by the inductive assumption, the expected number of queries that Player A need to make throughout the rest of the game (averaged over Player B's choices) is

$$p\left(d-1\right)2^{d/4+5}\sqrt{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^b))}. \tag{103}$$

There are two cases to consider. If $R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^b)) \le R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^{\bar{b}}))$, then combining Eq. (102) and Eq. (103), we have that the total number of queries averaged

over Player B's choices is

$$\begin{aligned}
&p\,(d-1)\,2^{(d-1)/4}\sqrt{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^b))}+p\,(d-1)\,2^{(d-1)/4+5}\sqrt{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^b))}\\
\leq&p\,(d-1)\,2^{(d-1)/4}\sqrt{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^b))}(1+2^5)\\
\leq&p\,(d-1)\,2^{(d-1)/4+5+1/16}\sqrt{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^b))}\\
\leq&p\,(d-1)\,2^{(d-1)/4+5+1/16+1/2}\sqrt{R_{s,t}(G_{\mathrm{NAND}_d}(x))}\\
\leq&p\,(d)\,2^{d/4+11/2}\sqrt{R_{s,t}(G_{\mathrm{NAND}_d}(x))}
\end{aligned}\tag{104}$$

where we've used $R_{s,t}(G_{\mathrm{NAND}_d}(x)) = \left(R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^0))^{-1}+R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^1))^{-1}\right)^{-1}$ from Claim 5 and the fact that $R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^b)) \leq R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^{\bar{b}}))$ to bound the value $R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^b))$ by $2R_{s,t}(G_{\mathrm{NAND}_d}(x))$. This proves the even induction step for this case.

The other case is if $R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^b)) > R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^{\bar{b}}))$. In that case, again using the fact that $R_{s,t}(G_{\mathrm{NAND}_d}(x)) = \left(R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^0))^{-1}+R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^1))^{-1}\right)^{-1}$, we have

$$R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^{\bar{b}})) = R_{s,t}(G_{\mathrm{NAND}_d}(x))\left(1+\frac{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^{\bar{b}}))}{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^b))}\right)^{-1} \leq \frac{2}{3}R_{s,t}(G_{\mathrm{NAND}_d}(x)),\tag{105}$$

where the inequality follows from Eq. (101). Thus, the average total number of queries is

$$\begin{aligned}
&p\,(d-1)\,2^{(d-1)/4}\sqrt{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^{\bar{b}}))}+p\,(d-1)\,2^{(d-1)/4+5}\sqrt{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^b))}\\
\leq&p(d-1)2^{(d-1)/4}\left(\sqrt{R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^{\bar{b}}))}+2^5\sqrt{2R_{s,t}(G_{\mathrm{NAND}_{d-1}}(x^{\bar{b}}))}\right)\\
\leq&p(d-1)2^{(d-1)/4}(1+2^{5+1/2})\sqrt{\frac{2}{3}R_{s,t}(G_{\mathrm{NAND}_d}(x))}\\
\leq&p(d)2^{d/4-1/4+5}\sqrt{R_{s,t}(G_{\mathrm{NAND}_d}(x))}\\
\leq&p(d)2^{d/4+5}\sqrt{R_{s,t}(G_{\mathrm{NAND}_d}(x))}.
\end{aligned}\tag{106}$$

This proves the induction step for the other case. □