Electronic Colloquium on Computational Complexity, Report No. 41 (2005)

(Almost) tight bounds for randomized and quantum Local Search on hypercubes and grids*

Shengyu Zhang†

Abstract

The Local Search problem, which finds a local minimum of a black-box function on a given graph, is of both practical and theoretical importance to many areas in computer science and natural sciences. In this paper, we show that for the Boolean hypercube $\{0,1\}^n$, the randomized query complexity of Local Search is $\Theta(2^{n/2}n^{1/2})$ and the quantum query complexity is $\Theta(2^{n/3}n^{1/6})$. We also show that for the constant dimensional grid $[N^{1/d}]^d$, the randomized query complexity is $\Theta(N^{1/2})$ for $d \geq 4$ and the quantum query complexity is $\Theta(N^{1/3})$ for $d \geq 6$. New lower bounds for lower dimensional grids are also given. These improve the previous results by Aaronson [1], and Santha and Szegedy[17]. Finally we show for $[N^{1/2}]^2$ a new upper bound of $O(N^{1/4}(\log\log N)^{3/2})$ on the quantum query complexity, which implies that Local Search on grids exhibits different properties at low dimensions.

1 Introduction

The Local Search problem on an undirected graph $G = (V, E)$ is defined as follows. Given a function $f : V \to \mathbb{N}$, find a vertex $v \in V$ such that $f(v) \leq f(w)$ for all neighbors w of v. Local Search has many connections to optimization theory and the total function problems in complexity theory. For example, the 2SAT-FLIP problem, an important problem known to be complete in the class **PLS** (Polynomial Local Search), is actually the Local Search problem on Boolean hypercube $\{0,1\}^n$, with $f(x)$ being the sum of the weights of the clauses that the truth assignment x satisfies. Here **PLS**, introduced by Johnson, Papadimitriou, and Yannakakis [11], is a subclass of **TFNP**, the family of total function problems introduced by Megiddo and Papadimitriou [15]. The Local Search problem is also related to physical systems including folding proteins, the quantum adiabatic algorithms, etc. See [1] and [17] for more discussions.

Given the theoretical and practical importance of this problem, there have been many works dedicated to it, especially in the query model [1, 2, 14, 13, 17]. In this model, $f(v)$ can only be accessed by querying v, and the randomized (and quantum) query complexity, denote by $RLS(G)$ (and $QLS(G)$) is the minimum number of queries needed by a randomized (and quantum) algorithm that solves the problem. Previously, for upper bounds on a general N-vertex graph G, Aldous [2] proved that $RLS(G) = O(\sqrt{N\delta})$ and Aaronson [1] proved that $QLS(G) = O(N^{1/3}\delta^{1/6})$, where δ is the maximum degree of G. For lower bounds, Aaronson [1] showed that for the Boolean hypercube $\{0,1\}^n$, $RLS(\{0,1\}^n) = \Omega(2^{n/2}/n^2)$ and $QLS(\{0,1\}^n) = \Omega(2^{n/4}/n)$, and that for the constant dimensional grid $[N^{1/d}]^d$, $RLS([N^{1/d}]^d) = \Omega(N^{1/2-1/d}/\log N)$ and $QLS([N^{1/d}]^d) = \Omega(N^{1/4-1/(2d)}/\sqrt{\log N})$. It has also been shown that $QLS([N^{1/2}]^2) = \Omega(N^{1/8})$ by Santha and Szegedy [17]. However, the final values of QLS and RLS on both types of graphs remain an open problem, explicitly stated in an earlier version of [1] and also (partially) in [17].

In this paper, we improve these previous results and show (almost) tight bounds on both RLS and QLS in a unified framework. For the Boolean hypercube, our lower bounds match the known upper bounds [1, 2]. For the constant dimensional grid graphs, our lower bounds also match the known upper bounds except for a few low dimensional cases.

Theorem 1 $RLS(\{0,1\}^n) = \Omega(2^{n/2}n^{1/2}), \quad QLS(\{0,1\}^n) = \Omega(2^{n/3}n^{1/6}).$

*This research was supported in part by NSF grant CCR-0310466.

†Computer Science Department, Princeton University, NJ 08544, USA. Email: szhang@cs.princeton.edu

ISSN 1433-8092

Theorem 2

$$RLS([N^{1/d}]^d) = \begin{cases} \Omega(N^{1/2}) & \text{if } d \geq 4 \\ \Omega((N/\log N)^{1/2}) & \text{if } d = 3 \\ \Omega\left(N^{1/3}\right) & \text{if } d = 2 \end{cases}, \quad QLS([N^{1/d}]^d) = \begin{cases} \Omega(N^{1/3}) & \text{if } d \geq 6 \\ \Omega((N/\log N)^{1/3}) & \text{if } d = 5 \\ \Omega\left(N^{1/2-1/(d+1)}\right) & \text{if } 2 \leq d \leq 4 \end{cases}$$

The proofs for the quantum lower bounds in both theorems use the quantum adversary method, which was originally proposed by Ambainis [4], and later generalized in different ways [3, 5, 12, 18]. Recently Spalek and Szegedy made the picture clear by showing that all these generalizations are equivalent in power [17]. On the other hand, in proving a particular problem, some of the methods might be easier to use than the others. In our case, the technique proposed by Zhang [18] works pretty well.

Inspired by the quantum adversary method, Aaronson gave a technique called rational adversary method, to prove lower bounds of randomized query complexity [1]. Our proofs for the randomized lower bounds will use this method.

Both the quantum adversary method and the rational adversary method are frameworks of proving lower bounds, parameterized by input sets and weight functions of input pairs. Both our proofs and Aaronson's proofs [1] use random walks in the corresponding graphs to give the input sets and weight functions. Besides choosing different random walks and different weight functions, a key innovation that distinguishes our work from Aaronson's is that we decompose the graph into two parts, the tensor product of which is the original graph. We perform the random walk only in one part, and perform a simple one-way walk in a self-avoiding path in the other part, which serves as a "clock" to record the number of steps taken by the random walk in the first part. The tensor product of these two walks is a random path in the original graph. A big advantage of adding a clock is that the "passing probability", the probability that the random path *passes* a vertex v *within* T steps, is now the "stopping probability", the probability that a random path *stops* at v *after* exactly t steps, which is well understood in the classical random walk literature. Another advantage is that since the walk in the second part is on a self-avoiding path, the resulting random path in the original graph does not intersect with itself either, which makes our analysis easier.

Finally, we give a new upper bound for $QLS([N^{1/2}]^2)$, which implies that Local Search on grids exhibits different properties at low dimensions.

Theorem 3 $QLS([N^{1/2}]^2) = O(N^{1/4}(\log\log N)^{3/2})$

Other related results. There were two unpublished results. It is mentioned in [1] that Ambainis showed $QLS(\{0,1\}^n) = \Omega(2^{n/3}/n^{O(1)})$, and it is mentioned in [17] that Verhoeven showed $RLS([N^{1/2}]^2) = \Omega(N^{1/2-\delta})$ for any constant $\delta > 0$.

2 Preliminaries and notations

We use $[M]$ to denote the set $\{1, 2, ..., M\}$. We define the sign function to be $sign(z) = 1$ if $z > 0$, -1 if $z < 0$ and 0 if $z = 0$. For an n-bit binary string $x = x_0...x_{n-1} \in \{0,1\}^n$, let $x^{(i)} = x_0...x_{i-1}(1-x_i)x_{i+1}...x_{n-1}$ be the string obtained by flipping the coordinate i.

A path X in a graph $G = (V, E)$ is a sequence $(v_1, ..., v_l)$ of vertices such that for any pair (v_i, v_{i+1}) of vertices, either $v_i = v_{i+1}$ or $(v_i, v_{i+1}) \in E$. We use $set(X)$ to denote the set of distinct vertices on path X.

The (k,l)-hypercube $G_{k,l}$ is a special graph whose vertex set is $V = [k]^l$ and whose edge set is $E = \{(u,v) : \exists i \in [l],\ s.t.\ |u_i - v_i| = 1, \text{ and } u_j = v_j,\ \forall j \neq i\}$. Sometimes we abuse the notation by using $[k]^l$ to denote $G_{k,l}$. Note that both the Boolean hypercube and the constant dimension grid are special hypercubes.[1]

In an N-vertex graph $G = (V, E)$, a Hamilton path is a path $X = (v_1, ..., v_{|V|})$ such that $(v_i, v_{i+1}) \in E$ for any $i \in [N-1]$ and $set(X) = V$. It is easy to check by induction that every hypercube $[k]^l$ has a Hamilton path. Actually, for $l = 1$, $[k]$ has a Hamilton path $(1, ..., k)$. Now suppose $[k]^l$ has a Hamilton path P, then a Hamilton path for $[k]^{l+1}$ can be constructed as follows,

[1] Here we identify the Boolean hypercube $\{0,1\}^n$ and $G_{2,n}$ since they are isomorphic.

first fix the last coordinate to be 1 and go through P, then change the last coordinate to be 2 and go through P in the reverse order, and change the last coordinate to be 3 and go through P, and so on. For each (k,l), let $HamPath_{k,l} = (v_1, ..., v_N)$ be the Hamilton path constructed as above, and we define the successor function $H_{k,l}(v_i) = v_{i+1}$ for $i \in [N-1]$.

We use $R_2(f)$ and $Q_2(f)$ to denote the double-sided error random and quantum query complexities of function f. For more details on query models and query complexities, we refer to [7] as an excellent survey.

2.1 One quantum adversary method and the relational adversary method

We describe the quantum adversary method proposed in [18]. The definition and theorem given here are a little more general than the original ones, but the proof remains unchanged.

Definition 1 *Let $F : I^N \to [M]$ be an N-variate function. Let $R \subseteq I^N \times I^N$ be a relation such that $F(x) \neq F(y)$ for any $(x,y) \in R$. A weight scheme consists of three weight functions $w(x,y) > 0$, $u(x,y,i) > 0$ and $v(x,y,i) > 0$ satisfying $u(x,y,i)v(x,y,i) \geq w^2(x,y)$ for all $(x,y) \in R$ and $i \in [N]$ with $x_i \neq y_i$. We further put*

$$w_x = \sum_{y':(x,y')\in R} w(x,y'), \qquad w_y = \sum_{x':(x',y)\in R} w(x',y) \tag{1}$$

$$u_{x,i} = \sum_{y':(x,y')\in R, x_i \neq y'_i} u(x,y',i), \qquad v_{y,i} = \sum_{x':(x',y)\in R, x'_i \neq y_i} v(x',y,i). \tag{2}$$

Theorem 4 [Zhang, [18]] *For any F, R and any weight scheme w, u, v as in Definition 1, we have*

$$Q_2(F) = \Omega\Big(\min_{(x,y)\in R, i\in[N], x_i\neq y_i} \sqrt{\frac{w_x w_y}{u_{x,i} v_{y,i}}}\Big) \tag{3}$$

In [1], Aaronson gives a nice technique to get a lower bound for randomized query complexity. We restate it using a similar language of Theorem 4.

Theorem 5 [Aaronson, [1]] *Let $F : I^N \to [M]$ be an N-variate function. Let $R \subseteq I^N \times I^N$ be a relation such that $F(x) \neq F(y)$ for any $(x,y) \in R$. For any weight function $w : R \to \mathbb{R}^+$, we have*

$$R_2(F) = \Omega\Big(\min_{(x,y)\in R, i\in[N], x_i\neq y_i} \max\{\frac{w_x}{w_{x,i}}, \frac{w_y}{w_{y,i}}\}\Big) \tag{4}$$

where

$$w_{x,i} = \sum_{y':(x,y')\in R, x_i\neq y'_i} w(x,y'), \qquad w_{y,i} = \sum_{x':(x',y)\in R, x'_i\neq y_i} w(x',y). \tag{5}$$

Note that we can think of Theorem 5 as also having a weight scheme, which requires that $u(x,y,i) = v(x,y,i) = w(x,y)$. This simple observation will be used in the proof of Theorem 1 and 2.

3 Lower bounds for Local Search on the Boolean Hypercube

The proof of Theorem 1 uses the following lemma. Consider that we put t balls randomly into m bins one by one. The j-th ball goes into the i_j-th bin. Denote by n_i the total number of balls in the i-th bin. We write $n_i \equiv b_i$ if $b_i = n_i \bmod 2$. We say that $(i_1, ..., i_t)$ *generates the parity sequence* $(b_1, ..., b_m)$, or simply $(i_1, ..., i_t)$ *generates* $(b_1, ..., b_m)$, if $n_i \equiv b_i$ for all $i \in [m]$. For $b_1...b_m \in \{0,1\}^m$, denote by $p^{(t)}[b_1, ..., b_m]$ the probability that $n_i \equiv b_i$, $\forall i \in [m]$. We may also require that the first ball is not put in the bin i^* for some $i^* \in [m]$. We use $p_{i^*}^{(t)}[b_1, ..., b_m]$ to denote the probability that $n_i \equiv b_i$, $\forall i \in [m]$, under the condition that the first ball is not put in the bin i^*. Let $p_i^{(t)} = \max_{b_1,...,b_m} p_i^{(t)}[b_1, ..., b_m]$ and $p_{i^*}^{(t)} = \max_{b_1,...,b_m} p_{i^*}^{(t)}[b_1, ..., b_m]$. The following bounds on $p_{i^*}^{(t)}$ are rather loose but sufficient for our purpose.

Lemma 6 *For any $i^* \in [m]$, we have*

$$p_{i^*}^{(t)} = \begin{cases} O(m^{-\lceil t/2 \rceil}) & \text{if } t \le 10 \\ O(m^{-5}) & \text{if } 10 < t \le m^2 \\ O(2^{-m}) & \text{if } t > m^2 \end{cases} \tag{6}$$

The proof of the lemma is in Appendix A. Now we are ready to prove Theorem 1.

Proof (of Theorem 1) We decompose the whole hypercube $\{0,1\}^n$ into two spaces V^w and V^c. The first space V^w is an m-dimensional hypercube $\{0,1\}^m$, where m, a fixed value only depending on d, will be given later. The second space V^c is an $(n-m)$-dimensional hypercube $\{0,1\}^{n-m}$. Obviously, $\{0,1\}^n = V^w \otimes V^c$, and each vertex $x = x_0...x_{n-1}$ in $\{0,1\}^n$ can be decomposed as $x = x^w \otimes x^c$ where $x^w = x_0...x_{m-1} \in V^w$ and $x^c = x_m...x_{n-1} \in V^c$. We shall use the two spaces in the following way. In V^w we perform a random walk; in V^c we set a "clock", recording how many steps the random walk in V^w has gone.

Consider the paths $X = (x_{0,0}, x_{0,1}, x_{1,0}, x_{1,1}, ..., x_{T,0}, x_{T,1})$ where $T = 2^{n-m} - 1$, that satisfies the following descriptions.

1. The starting point $x_{0,0} = x_{0,0}^w \otimes x_{0,0}^c$, where $x_{0,0}^w$ is any fixed point in V^w, say 00...0, and $x_{0,0}^c$ is the first vertex in the fixed Hamilton path $HamPath_{2,n-m}$ of V^c.
2. For each $t \in \{0, ..., T\}$,
   (a) $x_{t,1} = (x_{t,0}^w)^{(i_t^x)} \otimes x_{t,0}^c$, where $i_t^x \in \{0, ..., m-1\}$. That is, we randomly choose a coordinate i_t^x of $x_{t,0}^w$ and flip it.
   (b) $x_{t+1,0} = x_{t,1}^w \otimes H_{2,n-m}(x_{t,1}^c)$. That is, we let the clock "ticks" once.

Let the set P contain all such paths X's, then we define a problem $\textsc{Path}_P$: given a path $X \in P$, find the end point $x_{T,1}$. We are allowed to access X by querying an oracle O whether a point $x \in set(X)$ and getting the Yes/No answer. Note that an input of $\textsc{Path}_P$ is actually a Boolean function $g : \{0,1\}^n \to \{0,1\}$, with $g(x) = 1$ if and only if $x \in set(X)$. So strictly speaking, an input should be specified as $set(X)$ rather than X, because in general, it is possible that $X \neq Y$ but $set(X) = set(Y)$. For our problem, however, it is easy to check that for any $X, Y \in P$, we have $X = Y \Leftrightarrow set(X) = set(Y)$. (Actually, if $X \neq Y$, suppose the first diverging place is k, *i.e.* $x_{k,0} = y_{k,0}$, but $x_{k,1} \neq y_{k,1}$. Then X will never pass $y_{k,1}$ because the clock immediately ticks and the time always advances forward. Thus $set(X) \neq set(Y)$.) Therefore in what follows, we shall use X, Y... to specify inputs.

The following claim says that the $\textsc{Path}_P$ problem is no harder than the Local Search.

Claim 1 *$R_2(\textsc{Path}_P) \le 2RLS(\{0,1\}^n)$, $Q_2(\textsc{Path}_P) \le 2QLS(\{0,1\}^n)$.*

Proof For any path $X \in P$, we define a function f_X essentially in the same way as Aaronson did in [1]: for each $v \notin X$, $f_X(v) = \delta(v, x_{0,0}) + 2T$, where $\delta(u,v)$ is the Hamming distance between $u, v \in \{0,1\}^n$; for each $x_{t,b} \in set(X)$, $f_X(x_{t,b}) = 2(T-t) - b$. It is easy to check that the only local minimum point is $x_{T,1}$.

Suppose we have an Q-query randomized or quantum algorithm $\mathcal{A}$ for Local Search, we shall give a $2Q$ algorithm for $\textsc{Path}_P$. Given an oracle O and an input X of the $\textsc{Path}$ problem, we run $\mathcal{A}$ to find the local minimum point of f_X, which is also the end point of X. Whenever $\mathcal{A}$ needs to make a query on v to get $f_X(V)$, it asks O whether $v \in set(X)$. If $v \notin set(X)$, then $f_X(v) = \delta(v, x_{0,0}) + 2T$; otherwise, $v = x_{t,b}$ for some t and b (note that for a given $x_{t,b}$, t is fixed and known). If $t = 0$, then $f_X(v) = 2T$ if $v = x_{0,0}$ and $f_X(v) = 2T - 1$ otherwise. If $t > 0$, then we ask O whether $v^w \otimes H_{2,n}^{-1}(v^c) \in set(X)$. ($H_{2,n}^{-1}(v)$ gives the predecessor of v in the fixed Hamilton path.) If yes, then $v = x_{t,0}$ and thus $f_X(v) = 2(T-t)$; if no, then $v = x_{t,1}$ and $f_X(v) = 2(T-t) - 1$. Therefore, at most 2 queries on O can simulate one query on f, so we have a $2Q$ algorithm for $\textsc{Path}_P$. □

(Continue the proof of Theorem 1) By the claim, it is sufficient to prove lower bounds for $\textsc{Path}_P$. We define a relation R_P of paths as follows.

$$R_P = \{(X,Y) : X = (x_{0,0}, x_{0,1}, ..., x_{T,0}, x_{T,1}) \in P, Y = (y_{0,0}, y_{0,1}, ..., y_{T,0}, y_{T,1}) \in P, x_{T,1} \neq y_{T,1}\} \tag{7}$$

We then choose the weight functions. Recall that for a path X, i^x_t is the coordinate flipped at time t. For any $(X,Y)\in R_P$, we write $X\wedge Y=k$ if $i^x_0=i^y_0$, ..., $i^x_{k-1}=i^y_{k-1}$ but $i^x_k\neq i^y_k$. Let

$$w(X,Y)=1/|\{Z\in P: Z\wedge X=k\}|, \tag{8}$$

Now let us calculate w_X. By definition, $w_X=\sum_{Y':(X,Y')\in R_P} w(X,Y')$. We group those Y' that diverge from X at the same place. Then

$$\begin{aligned} w_X &= \sum_{k=0}^{T}\sum_{Y':(X,Y')\in R_P, X\wedge Y'=k} w(X,Y') &(9)\\ &= \sum_{k=0}^{T}\sum_{Y':(X,Y')\in R_P, X\wedge Y'=k}\frac{1}{|\{Z\in P: Z\wedge X=k\}|} &(10)\\ &= \sum_{k=0}^{T}\mathbf{Pr}_{Y'\in P}[y'_{T,1}\neq x_{T,1}|Y'\wedge X=k] &(11)\end{aligned}$$

By definition, if $Y'\wedge X=T$, then $y'_{T,1}\neq x_{T,1}$ for sure. If $k<T$, note that for those Y' that $Y'\wedge X=k$, $y'_{T,1}=x_{T,1}$ if and only if $(i^{y'}_k,...,i^{y'}_T)$ generates the same parity sequence $(b_1,...,b_m)$ as $(i^x_k,...,i^x_T)$ does. Thus $\mathbf{Pr}_{Y'\in P}[y'_{T,1}\neq x_{T,1}|Y'\wedge X=k]=1-p^{(T-k+1)}_{i^x_k}[b_1,...,b_m]=1-o(1)$ by Lemma 6. It follows that $w_X=\sum_{k=0}^{T-1}(1-p^{(T-k+1)}_{(i^x_k)}[b_1,...,b_m])+1=T-o(T)$. Similarly, we have also $w_Y=T-o(T)$.

Now we define $u(X,Y,i)$ and $v(X,Y,i)$, where i is a point $x_{j,b}\in set(X)-set(Y)$ or $y_{j,b}\in set(Y)-set(X)$.

$$u(X,Y,x_{j,b})=a_{k,j,b}w(X,Y),\qquad u(X,Y,y_{j,b})=b_{k,j,b}w(X,Y), \tag{12}$$

$$v(X,Y,x_{j,b})=b_{k,j,b}w(X,Y),\qquad v(X,Y,y_{j,b})=a_{k,j,b}w(X,Y). \tag{13}$$

where $a_{k,j,b}b_{k,j,b}=1$, and the values of $a_{k,j,b}$ and $b_{k,j,b}$ will given later. We now calculate $u_{X,i}$ and $v_{Y,i}$ for $i=x_{j,b}\in set(X)-set(Y)$; the other case $i=y_{j,b}$ is just symmetric. Note that since $x_{j,b}\in set(X)-set(Y)$, we have $k\leq j-1$ if $b=0$ and $k\leq j$ if $b=1$.

$$\begin{aligned} u_{X,x_{j,b}} &= \sum_{k=0}^{j+b-1}\sum_{Y':(X,Y')\in R,\ X\wedge Y'=k,\ x_{j,b}\notin set(Y')} a_{k,j,b}w(X,Y') &(14)\\ &\leq \sum_{k=0}^{j+b-1}\sum_{Y'\in P: X\wedge Y'=k} a_{k,j,b}w(X,Y')=\sum_{k=0}^{j+b-1}a_{k,j,b} &(15)\end{aligned}$$

The computation for $v_{Y,x_{j,b}}$ is a little more complicated. By definition,

$$\begin{aligned} v_{Y,x_{j,b}} &= \sum_{k=0}^{j+b-1}\sum_{X':(X',Y)\in R,\ X'\wedge Y=k,\ x_{j,b}\in set(X')} b_{k,j,b}w(X',Y) &(16)\\ &\leq \sum_{k=0}^{j+b-1}\sum_{X'\in P: X'\wedge Y=k,\ x_{j,b}\in set(X')} b_{k,j,b}w(X',Y) &(17)\\ &= \sum_{k=0}^{j+b-1} b_{k,j,b}\mathbf{Pr}_{X'\in P}[x_{j,b}\in set(X')|X'\wedge Y=k] &(18)\end{aligned}$$

Note that because of the clock, $x_{j,b}\in set(X')$ if and only if $x_{j,b}=x'_{j,b'}$ for some $b'\in\{0,1\}$. And actually $b=b'$, because otherwise $x_{j,b}$ and $x_{j,b'}$ have different parities of number of 1's. Therefore, $\mathbf{Pr}_{X'}[x_{j,b}\in set(X')|X'\wedge Y=k]=\mathbf{Pr}_{X'}[x_{j,b}=x'_{j,b}|X'\wedge Y=k]=p^{(j-k+b)}_{i^y_k}[b_1,...,b_m]\leq p^{(j-k+b)}_{i^y_k}$, where $(b_1,...,b_m)$ is the parity sequence generated by $i^x_k,...,i^x_{j+b-1}$. So

$$v_{Y,x_{j,b}} \le \sum_{k=0}^{j+b-1} b_{k,j,b} p^{(j-k+b)}_{i^y_k} = O(\sum_{k=0}^{j-m^2+b-1} b_{k,j,b}/2^m + \sum_{k=j-m^2+b}^{j+b-11} b_{k,j,b}/m^5 + \sum_{k=j+b-10}^{j+b-1} b_{k,j,b}/m^{\lceil (j-k+b)/2\rceil}) \tag{19}$$

Now for the randomized lower bound purpose, we pick $m = \lfloor (n+\log_2 n)/2 \rfloor$, $a_{k,j,b} = b_{k,j,b} = 1$. Then $T = 2^{n-m} - 1 = \Theta(2^{n/2}/\sqrt{n})$, $\frac{w_X}{u_{X,x_{j,b}}} = \frac{T-o(T)}{j} \ge 1 - o(1)$, and

$$\frac{w_Y}{v_{Y,x_{j,b}}} = \Omega\left(\frac{T-o(T)}{\frac{j}{2^m} + m^2/m^5 + \sum_{t=1}^{5} 1/m^t}\right) = \Omega\left(\frac{2^{n/2}/\sqrt{n}}{\frac{2^{n/2}/\sqrt{n}}{\sqrt{n}2^{n/2}} + 1/n}\right) = \Omega(\sqrt{n}2^{n/2}). \tag{20}$$

It is easy to check using the same calculations that for any $y_{j,b} \in set(Y) - set(X)$, $\frac{w_X}{u_{X,y_{j,b}}} = \Omega(\sqrt{n}2^{n/2})$, and $\frac{w_Y}{v_{Y,y_{j,b}}} \ge 1 - o(1)$. Therefore, in either case ($i = x_{j,b}$ or $i = y_{j,b}$), we have

$$RLS(\{0,1\}^n) = \max\{\frac{w_X}{u_{X,i}}, \frac{w_Y}{v_{Y,i}}\} = \Omega(\sqrt{n}2^{n/2}) \tag{21}$$

For the quantum lower bound, we pick $m = \lfloor (2n - \log n)/3 \rfloor$, and

$$a_{k,j,b} = \begin{cases} m^{-\lceil (j-k+b)/2\rceil/2} & \text{if } j-k+b \le 10 \\ m^{-5/2} & \text{if } 10 < j-k+b \le m^2 \\ 2^{-m/2} & \text{if } j-k+b > m^2 \end{cases}, \quad b_{k,j,b} = \begin{cases} m^{\lceil (j-k+b)/2\rceil/2} & \text{if } j-k+b \le 10 \\ m^{5/2} & \text{if } 10 < j-k+b \le m^2 \\ 2^{m/2} & \text{if } j-k+b > m^2 \end{cases} \tag{22}$$

Clearly $a_{k,j,b}b_{k,j,b} = 1$ holds. Note that $T = 2^{n-m} - 1 = \Theta(2^{n/3}n^{2/3})$. Thus $w_X = w_Y = \Omega(T) = \Omega(2^{n/3}n^{2/3})$, and

$$u_{X,x_{j,b}} \le \sum_{k=0}^{j+b-1} a_{k,j,b} = \sum_{k=0}^{j-m^2+b-1} 2^{-m/2} + \sum_{k=j-m^2+b}^{j+b-11} m^{-5/2} + \sum_{k=j+b-10}^{j+b-1} m^{-\lceil (j-k+b)/2\rceil/2} = O(\sqrt{n}) \tag{23}$$

$$v_{Y,x_{j,b}} \le O(\sum_{k=0}^{j-m^2+b-1} 2^{m/2}/2^m + \sum_{k=j-m^2+b}^{j+b-11} m^{5/2}/m^5 + \sum_{k=j+b-10}^{j+b-1} m^{\lceil (j-k-1)/2\rceil/2}/2^{\lceil (j-k-1)/2\rceil}) = O(\sqrt{n}) \tag{24}$$

It is easy to check that the above inequalities all hold for the symmetric case of $y_{j,b}$, so

$$QLS(\{0,1\}^n) = \Omega\left(\sqrt{\frac{(2^{n/3}n^{2/3})(2^{n/3}n^{2/3})}{O(\sqrt{n})O(\sqrt{n})}}\right) = \Omega(2^{n/3}n^{1/6}). \tag{25}$$

□

4 Lower bounds for Local Search on the constant dimensional grid

To simplify notations, we let $n = N^{1/d}$. For $x = x_0...x_{d-1}$ in $[n]^d$, let $x^{(k)=l} = x_0...x_{k-1}lx_{k+1}...x_{d-1}$, and $x^{(k)=(k)+i} = x_0...x_{k-1}(x_k+i)x_{k+1}...x_{d-1}$, where i satisfies $x_k + i \in [n]$. Also let $x^{(i),-} = x^{(i)=\max\{x_i-1,1\}}$ and $x^{(i),+} = x^{(i)=\min\{x_i+1,n\}}$.

4.1 1-dimensional short walk

We will use random walk on an n-point line, where a particle is initially put at point $i \in \{1, ..., n\}$, and in each step the particle moves either to $\max\{1, i-1\}$ or to $\min\{n, i+1\}$ with equal probability. That is, the particle randomly choose to move left or right, but if it is currently at the left (or right) end and still wants to move left (or right), then it stands still. We refer to it as short walk. Let $p_{ij}^{(t)}$ denote the probability that the particle starting from point i stops at point j after exact t steps of the walk. Obviously, we have $\max_{i,j} p_{ij}^{(t)} = 1$ if $t = 0$. For $t \geq 1$, the following proposition gives a good estimate on $\max_{ij} p_{ij}^{(t)}$.

Proposition 7 *For any $t \geq 1$,*

$$\max_{i,j} p_{ij}^{(t)} = \begin{cases} O(1/\sqrt{t}) & \text{if} \quad t \leq n^2 \\ O(1/n) & \text{if} \quad t > n^2 \end{cases} \tag{26}$$

The proof of the proposition is in Appendix B.

4.2 Weaker lower bounds

We shall first show a weaker result in this section, then we improve it in section 4.3. As in the proof of Theorem 1, we decompose the space $[n]^d$ into two parts $V^w \otimes V^c$, where $V^w = [n]^m$ and $V^c = [n]^c$. Each vertex $x = x_0...x_{d-1}$ in $[n]^d$ can be decomposed as $x = x^w \otimes x^c$ where $x^w = x_0...x_{m-1} \in V^w$ and $x^c = x_m...x_{d-1} \in V^c$. Consider the paths $X = (x_{0,0}, x_{0,1}, x_{1,0}, x_{1,1}, ..., x_{T,0}, x_{T,1})$, where $T = n^{d-m} - 1$, satisfying the following description.

1. The starting point $x_{0,0} = x_{0,0}^w \otimes x_{0,0}^c$, where all coordinates of $x_{0,0}^w$ are $\lfloor n/2 \rfloor$, and $x_{0,0}^c$ is the first vertex in the fixed Hamilton path $HamPath_{n,d-m}$ of V^c.
2. For each $t \in \{0, ..., T\}$,
   (a) $x_{t,1} \in \{x_{t,0}^{(t \bmod m),+}, x_{t,0}^{(t \bmod m),-}\}$.
   (b) $x_{t+1,0} = x_{t,1}^w \otimes H(x_{t,1}^c)$.

Let P contain all such paths X's, then we define the Path_P problem in the same way as in the proof of Theorem 1, and it is easy to show that $R_2(\text{Path}_P) \leq 2RLS([n]^d)$ and $Q_2(\text{Path}_P) \leq 2QLS([n]^d)$. We write $X \wedge Y = k$ if $x_{0,0} = y_{0,0}$, $x_{0,1} = y_{0,1}$, ..., $x_{k,0} = y_{k,0}$ but $x_{k,1} \neq y_{k,1}$. We then define R_P and all weight functions w, u, v in the same form as those in the proof of Theorem 1 (*i.e.* (7)(8)(12)(13)). For two points $z_1, z_2 \in V^w$, define $z_1 \rightarrow_t^l z_2$ to be the event that a random walk starting at z_1 stops at z_2 after exact t steps, performing one step of short walk in dimension $((l + s - 1) \bmod m)$ in the s-th step ($s \in [t]$). By Proposition 7, we know that $\mathbf{Pr}[z_1 \rightarrow_t^l z_2] = O(\frac{1}{\sqrt{t/m}^m}) = O(\frac{1}{\sqrt{t}^m})$ if $1 \leq t \leq mn^2$, and $\mathbf{Pr}[z_1 \rightarrow_t^l z_2] = O(1/n^m)$ if $t > mn^2$. By some calculations similar to those in the proof of Theorem 1, we have $w_X = w_Y = T - o(T)$, $u_{X,x_{j,b}} \leq \sum_{k=0}^{j+b-1} a_{k,j,b}$, and $v_{Y,x_{j,b}} \leq \sum_{k=0}^{j+b-1} b_{k,j,b}\mathbf{Pr}_{X'}[x_{j,b} \in set(X') | X' \wedge Y = k]$. Note that $x_{j,b} \in set(X') \Leftrightarrow x_{j,b} = x'_{j,b}$ again due to the clock and the parity. Also note that if $X' \wedge Y = k$, then $x_{j,b} = x'_{j,b} \Leftrightarrow x_{k,1}^w \rightarrow_{j-k+b-1}^{(k+1) \bmod m} x_{j,b}^w$. Therefore,

$$v_{Y,x_{j,b}} \leq \sum_{k=0}^{j+b-1} b_{k,j,b}\mathbf{Pr}[x_{k,1}^w \rightarrow_{j-k+b-1}^{(k+1) \bmod m} x_{j,b}^w] \tag{27}$$

$$= O\left(\sum_{k=0}^{j-mn^2+b-1} \frac{b_{k,j,b}}{n^m} + \sum_{k=j-mn^2+b}^{j+b-2} \frac{b_{k,j,b}}{\sqrt{(j-k+b-1)^m}} + b_{j+b-1,j,b} \right) \tag{28}$$

Now for the randomized lower bound purpose, we take $a_{k,j,b} = b_{k,j,b} = 1$. Then $\frac{w_X}{u_{X,x_{j,b}}} = \Omega(1)$, and $v_{Y,x_{j,b}} = O\left(\frac{T}{n^m} + \sum_{i=1}^{mn^2} i^{-m/2} + 1\right) = O\left(n^{d-2m} + \sum_{i=1}^{mn^2} i^{-m/2} + 1\right)$. When $d > 4$ we pick $m = \lceil d/2 \rceil > 2$, then $v_{Y,x_{j,b}} = O(n^{d-2m}) + O(1)$ and $\frac{w_Y}{v_{Y,x_{j,b}}} = \Omega(n^{\lfloor d/2 \rfloor})$. Therefore

$$RLS([n]^d) = \Omega(R_2(\text{Path}_P)) = \Omega(\max\{\frac{w_X}{u_{X,x_{j,b}}}, \frac{w_Y}{v_{Y,x_{j,b}}}\}) = \begin{cases} \Omega(N^{\frac{1}{2}}) & \text{if } d = 2d' \\ \Omega(N^{\frac{1}{2}-\frac{1}{2d}}) & \text{if } d = 2d'+1 \end{cases} \tag{29}$$

For $d = 4$ and 3, we let $m = 2$ and get $RLS([n]^4) = \Omega(n^2/\log n) = \Omega(N^{1/2}/\log N)$ and $RLS([n]^3) = \Omega(n) = \Omega(N^{1/3})$. For $d = 2$, we let $m = 1$ and note that now the walk has only n long, so $w_Y = \Theta(n)$, $v_{Y,x_{j,b}} = O(\sqrt{n})$, and so $RLS([n]^2) = \Omega(\sqrt{n}) = \Omega(N^{1/4})$.

For the quantum lower bounds, take

$$a_{k,j,b} = \begin{cases} 1 & \text{if } j-k+b = 1 \\ (j-k+b-1)^{-m/4} & \text{if } 1 < j-k+b \le mn^2 \\ n^{-m/2} & \text{if } j-k+b > mn^2 \end{cases} \quad b_{k,j,b} = \begin{cases} 1 & \text{if } j-k+b = 1 \\ (j-k+b-1)^{m/4} & \text{if } 1 < j-k+b \le mn^2 \\ n^{m/2} & \text{if } j-k+b > mn^2 \end{cases} \tag{30}$$

Then $u_{X,x_{j,b}} = v_{Y,x_{j,b}} = O\left(Tn^{-m/2} + \sum_{i=1}^{n^2} i^{-m/4}\right)$, and

$$QLS([n]^d) = \Omega(Q_2(\text{PATH}_P)) = \Omega(\sqrt{\frac{w_X w_Y}{u_{X,x_{j,b}} v_{Y,x_{j,b}}}}) = \Omega\left(\frac{T}{Tn^{-m/2} + \sum_{i=1}^{n^2} i^{-m/4}}\right) \tag{31}$$

If $d > 6$, then we let m be the integer closest to $2d/3$, thus $m > 4$. We get

$$QLS([n]^d) = \begin{cases} \Omega(N^{\frac{1}{3}}) & \text{if } d = 3d' \\ \Omega(N^{\frac{1}{3}-\frac{1}{3d}}) & \text{if } d = 3d'+1 \\ \Omega(N^{\frac{1}{3}-\frac{1}{6d}}) & \text{if } d = 3d'+2 \end{cases}. \tag{32}$$

For $d = 6$, let $m = 4$ and we have $QLS([n]^6) = \Omega(n^2/\log n) = \Omega(N^{1/3}/\log N)$. For $d = 5, 4, 3$, we let $m = d - 2$ and then $w_Y = \Theta(n^2)$, $v_{Y,x_{j,b}} = O(n^{3-d/2})$ and $QLS([n]^d) = \Omega(n^{d/2-1}) = \Omega(N^{1/2-1/d})$. For $d = 2$, let $m = 1$ and $QLS([n]^2) = \Omega(\frac{n}{n^{3/4}}) = \Omega(n^{1/4}) = \Omega(N^{1/8})$.

4.3 Improvement

One weakness of the above proof is the integer constraint of the dimension m. We now show a way to avoid the problem. The idea is to partition the grid into many blocks, and different blocks represent different time slots.

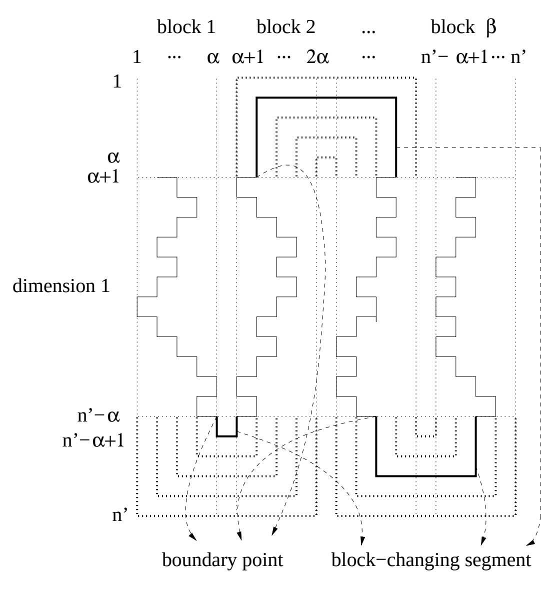

Figure 1: Illustration for changing a block in 2 dimensional grid

For any fixed r, where r will be given later, let $\alpha = \lfloor n^r \rfloor$, $\beta = \lfloor n^{1-r} \rfloor$ and $n' = \alpha\beta$. We now consider the slightly smaller grid $[n']^d$. Let V_1 be the set $\{x_0 ... x_{d-2} : x_i \in [n']\}$. We cut V_1 into

β^{d-1} parts, each of which is a small grid $[\alpha]^{d-1}$. We refer to the set $\{x_0...x_{d-2}x_{d-1} : (k_i-1)\alpha < x_i \le k_i\alpha, i = 0, ..., d-2, \alpha < x_{d-1} \le n' - \alpha\}$ as the *block* $(k_0, ..., k_{d-2})$. Note that $(k_0, ..., k_{d-2})$ can be also viewed as a point in grid $[\beta]^{d-1}$, and there is a Hamilton path $HamPath_{\beta,d-1}$ in $[\beta]^{d-1}$, as defined in Section 2. We call the block $(k'_0, ..., k'_{d-2})$ *the next block* of the block $(k_0, ..., k_{d-2})$ if $(k'_0, ..., k'_{d-2})$, viewed as the point in $[\beta]^{d-1}$, is the next point of $(k_0, ..., k_{d-2})$ in $HamPath_{\beta,d-1}$. Note that in $HamPath_{\beta,d-1}$, to go to the point next to $(k_0, ..., k_{d-2})$, only one of $k_0, ..., k_{d-2}$ changes by increasing or decreasing by 1. We call the the block $(k_0, ..., k_{d-2})$ *the last block* if $(k_0, ..., k_{d-2})$ is the last point in $HamPath_{\beta,d-1}$.

Now we define the random walk by describing how a particle may go from start to end. The path set is just all the possible paths the particle goes along. Intuitively, within one block, the last dimension $d-1$ is the clock space as before. If we run out of it, we say we reach *a boundary point* at the current block, and we change to the next block via a path segment called *block-changing segment*. In what follows, we always use $x_0...x_{d-1}$ to denote the current position of the particle. Thus $x_0 = x_0 + 1$, for example, means the particle moves from $x_0...x_{d-1}$ to $(x_0+1)x_1...x_{d-1}$. We also use $(k_0, ..., k_{d-2})$ to denote the current block which the particle is in.

1. Initially $x_0 = ... = x_{d-2} = \lfloor \alpha/2 \rfloor$, $x_{d-1} = \alpha + 1$, $k_0 = ... = k_{d-2} = 1$.
2. **for** $t = 0$ **to** $(n' - 2\alpha)\beta^{d-1} - 1$,

    Let $t' = \lfloor \frac{t}{n'-2\alpha} \rfloor$, $i = t \bmod (d-1)$

    do either $x_i = \max\{x_i - 1, (k_i - 1)\alpha + 1\}$ or $x_i = \min\{x_i + 1, k_i\alpha\}$ randomly

    if $t \ne k(n' - 2\alpha) - 1$ for some positive integer k,

      $x_{d-1} = x_{d-1} + (-1)^{t'}$

    else

      if the particle is not in the last block

      (Suppose the current block changes to the next block by increasing k_j by $b \in \{-1, 1\}$)

        do $x_{d-1} = x_{d-1} + (-1)^{t'}$ **for** $(\alpha + 1 - x_j)$ times

        do $x_j = x_j + b$ **for** $2(\alpha + 1 - x_j) - 1$ times

        do $x_{d-1} = x_{d-1} + (-1)^{t'+1}$ **for** $(\alpha + 1 - x_j)$ times

        $k_j = k_j + b$

      else

        The particle stops and the random walk ends

It is easy to check that every boundary point has one unique block-changing segment, and different block-changing segments do not intersect. Thus the block-changing segments thread all the blocks to form a $[\alpha]^{d-1} \times [L]$ grid, where $L = (n' - 2\alpha)\beta^{d-1}$. Actually it is not hard to check that for the proof of the lower bound purpose, we can just think of the new path set as being defined in the $[\alpha]^{d-1} \times [L]$ grid as in Section 4.2, with $V^w = [\alpha]^{d-1}$ and $V^c = [L]$. [2] So we have $w_X = w_Y = \Omega(T)$, where $T = L - 1 = \Theta(n^{1+(d-1)(1-r)})$. Also it holds $u_{X,x_{j,b}} \le O(\sum_{k=0}^{j+b-1} a_{k,j,b})$ and $v_{Y,x_{j,b}} = O\left(\sum_{k=0}^{j-(d-1)\alpha^2+b-1} \frac{b_{k,j,b}}{\alpha^{d-1}} + \sum_{k=j-(d-1)\alpha^2+b}^{j+b-2} \frac{b_{k,j,b}}{(j-k+b-1)^{-(d-1)/2}} + b_{j+b-1,j,b}\right)$.

For randomized lower bound, $a_{k,j,b} = b_{k,j,b} = 1$, $v_{Y,x_{j,b}} = O(T/\alpha^{d-1} + \sum_{t=1}^{(d-1)\alpha^2} t^{-(d-1)/2})$. So $w_Y/v_{Y,x_{j,b}} = \Omega\left(\min\left\{n^{(d-1)r}, \frac{n^{1+(d-1)(1-r)}}{\sum_{t=1}^{(d-1)\lfloor n^r \rfloor^2} t^{-(d-1)/2}}\right\}\right)$ by noting that $\alpha = \Theta(n^r)$ and $\beta = \Theta(n^{1-r})$. If $d \ge 4$, then let $r = d/(2d-2)$ and we get $RLS([n]^d) = \Omega(N^{1/2})$. If $d = 3$, let $r = 3/4 - \log\log n/(4\log n)$, and we get $RLS([n]^3) = \Omega(N^{1/2}/\sqrt{\log N})$. For $d = 2$, let $r = 2/3$ and we get $RLS([n]^2) = \Omega(N^{1/3})$.

For the quantum lower bounds, u, v are defined as in (12) and (13), where $a_{k,j,b}$ is equal to 1 if $j + b - k = 1$, equal to $(j - k + b - 1)^{-(d-1)/4}$ if $1 < j - k + b \le (d-1)\alpha^2$, and equal to $\alpha^{-(d-1)/2}$ if $j-k+b > (d-1)\alpha^2$, and $b_{k,j,b} = a_{k,j,b}^{-1}$. Then $QLS([n]^d) = \Omega\left(\min\left\{n^{(d-1)r/2}, \frac{n^{1+(d-1)(1-r)}}{\sum_{t=1}^{(d-1)\lfloor n^r \rfloor^2} t^{-\frac{d-1}{4}}}\right\}\right)$. Now if $d \ge 6$, then letting $r = 2d/(3d-3)$ and we get $QLS([n]^d) = N^{1/3}$. If $d = 5$, then let $r = 5/6 - \log\log n/(6\log n)$ and $QLS([n]^5) = (N/\log N)^{1/3}$. For $2 \le d \le 5$, we let $r = d/(d+1)$, then $QLS([n]^d) = N^{1/2-1/(d+1)}$. This completes the proof of Theorem 2.

[2] See Appendix C for more explanations.

5 The new upper bound on the 2-dimensional grid

In [1], a quantum algorithm for Local Search on general graphs is given as follows. Do a random sampling over all the vertices, find the minimum f-value vertex v in them using a result by Durr and Hoyer [9] based on Grover search [10]. If v is a local minimal vertex, then return v; otherwise we follow a *decreasing path* as follows. Find a neighbor of v with the smallest f-value, and continue this minimum-value-neighbor search process until getting to a local minimum vertex. Here our idea is that after finding the minimum vertex of the sampled points, in stead of following the decreasing path, we start over within a smaller grid and do this recursively. The precise algorithm is as follows.

1. $v^{(0)} = (\lfloor n/2 \rfloor, \lfloor n/2 \rfloor)$.
2. For $i = 1$ to $T = \frac{1}{2}\log n$ do
    (a) Pick a set $V^{(i)}$ of $2^{3-i} n \log(2\log n)$ random points in $U^{(i)} = \{(x_1, x_2) \in [n]^2 : |x_1 - (v^{(i-1)})_1| \leq n/2^i, |x_2 - (v^{(i-1)})_2| \leq n/2^i\}$.
    (b) Find one $v^{(i)}$ *s.t.* $f(v^{(i)}) = \min_{v \in V^{(i)}} f(v)$ using Durr and Hoyer's algorithm [9]. Repeat the mininum-finding for $\log(2\log n)$ steps to make the error probability less than $1/2\log n$.
3. Find u *s.t.* $f(u) = \min_{v \in U^{(T)}} f(v)$.

It is easy to show inductively that at each iteration i, with probability at least $1 - 1/2\log n$, $|\{v \in U^{(i)} : f(v) < f(v^{(i)})\}| \leq n/2^{i+1}$. So the decreasing path starting from $v^{(i)}$ is at most $n/2^{i+1}$ long. Therefore, there must be a locally minimum point in $U^{(i+1)}$ (for the next iteration to search). Finally $U^{(T)}$ is of $\Theta(\sqrt{n}) \times \Theta(\sqrt{n})$ size, so a direct minimum-finding will do.

Now let us analyze the error probability. There are two types of errors. One is that at some iteration i, $|\{v \in U^{(i)} : f(v) < f(v^{(i)})\}| > n/2^{i+1}$. The probability that it happens in iteration i is less than $1/2\log n$, so the whole probability that this type of errors happens at some iteration is less than $T/2\log n = 1/4$. Another type of errors is the minimum search, *i.e.* step 2(b), and step 3. For step 2(b), by repeating the minimum-finding we have let the error probability less than $1/2\log n$, so the probability that this error happens at some iteration is less than $T/2\log n = 1/4$. For step 3, we can let the error probability less than $1/4$. So the whole error probability is less than $3/4$.

The number of queries needed is $\sum_{i=1}^{\frac{1}{2}\log n} O(\log(2\log n)\sqrt{2^{3-i} n \log(2\log n)}) + O(\sqrt{\sqrt{n} \times \sqrt{n}}) = O(\sqrt{n}(\log\log n)^{1.5})$.

6 Concluding Remarks

The paper gives new lower bounds for Local Search problems. Some other random walk can be used to further improve the lower bound on low dimension grid cases. For example, by cutting the 2-dimensional grid into $n^{2/5}$ blocks (each of size $n^{4/5} \times n^{4/5}$) and using a random walk similar to Aaronson's in [1] (but with some modifications to make the path self-avoiding), we can prove $QLS([n]^2) = N^{1/5}/\log N$. But this walk suffers from the fact that the "passing probability" is now $n^{4/5}$ times the "stopping probability". So it only works better at dimension 2. We put the further results in a complete version of the paper.

The lower bound technique we use can be easily generalized to the Local Search on product graphs. Precisely, $G = (V, E)$ is a product graph if G can be decomposed as $G_1 \times G_2 = (V_1 \times V_2, (E_1 \times I_2) \cup (I_1 \times E_2))$ where $I_i = \{(v_i, v_i) : v_i \in V_i\}$ for $i = 1, 2$. Some graphs, like hypercubes, may have many ways of decomposition. For a fixed decomposition $\mathcal{D}$, suppose we have a random walk W on graph $G_1 = (V_1, E_1)$ with transition probability $\{p_{ij} : i = j \text{ or } (i,j) \in E\}$ and stationary distribution π. Denote by $p_{ij}^{(t)}$ the probability that the random walk starting at i stops at j after t steps. Let $p^{(t)} = \max_{ij} p_{ij}^{(t)}$ and $\pi_{max} = \max_{i \in V_1} \pi(i)$. We say the walk mixes at time t_0 if $p^{(t_0)} \leq 2\pi_{max}$. Let $p_1 = \sum_{t \leq t_0} p^{(t)}$ and $p_{1/2} = \sum_{t \leq t_0} \sqrt{p^{(t)}}$. Then under mild conditions, we have

$$RLS(G) = \Omega(\max_{\mathcal{D}} \min\{\frac{1}{\pi_{max}}, \frac{L}{p_1}\}), \qquad QLS(G) = \Omega(\max_{\mathcal{D}} \min\{\frac{1}{\sqrt{\pi_{max}}}, \frac{L}{p_{1/2}}\}) \tag{33}$$

where L is the length of the longest self-avoiding path in G_2.

Random walk has been widely studied as a sampling method for algorithms, where the key parameter is the mixing time. It is interesting that both Aaronson's [1] and this paper use random walk to give lower bounds. And we can see from (33) that for lower bounds, we care not only about the mixing time of the random walk, but also about its behavior before mixing.

Acknowledgement

The author thanks Scott Aaronson, Xiaoming Sun and Andy Yao very much for many valuable discussions.

References

[1] S. Aaronson. Lower Bounds for Local Search by Quantum Arguments, Proceedings of the thirty-sixth Annual ACM Symposium on Theory of Computing, pp. 465-474, 2004.

[2] D. Aldous. Minimization algorithms and random walk on the d-cube, Annals of Probability, 11(2), pp.403-413, 1983.

[3] A. Ambainis. Polynomial degree vs. quantum query complexity. Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science, pp. 230-239, 2003.

[4] A. Ambainis. Quantum lower bounds by quantum arguments, Journal of Computer and System Sciences, 64, pp. 750-767, 2002.

[5] H. Barnum, M. Saks, M. Szegedy. Quantum query complexity and semidefinite programming. Proceedings of the 18th Annual IEEE Conference on Computational Complexity, pp. 179-193, 2003.

[6] R. Beals, H. Buhrman, R. Cleve, M.Mosca, R. deWolf. Quantum lower bounds by polynomials. Journal of ACM, 48, pp. 778-797, 2001.

[7] H. Buhrman, R. de Wolf. Complexity measures and decision tree complexity: a survey. Theoretical Computer Science, Volume 288, Issue 1, pp. 21-43, 2002.

[8] C. Durr, M. Heiligman, P. Hoyer, M. Mhalla. Quantum query complexity of some graph problems. Proceedings of the 31st International Colloquium on Automata, Lanaguages, and Programming, pp. 481-493, 2004.

[9] C. Durr, P. Hoyer. A quantum algorithm for finding the minimum, 1996. quant-ph/9607014

[10] L. Grover. A fast quantum mechanical algorithm for database search, Proceedings of the 28th Annual ACM Symposium on the Theory of Computing, pp. 212-219, 1996.

[11] D. Johnson, C. Papadimitriou, and M. Yannakakis. How easy is local search, Journal of Computer and System Sciences 37, pp. 429448, 1988.

[12] S. Laplante, F. Magniez. Lower bounds for randomized and quantum query complexity using Kolmogorov arguments, Proceedings of the 19th Annual IEEE Conference on Computational Complexity, pp. 294-304, 2004.

[13] D. Llewellyn and C. Tovey. Dividing and conquering the square. Discrete Applied Mathematics 43, pp. 131-153, 1993.

[14] D. Llewellyn, C. Tovey. and M. Trick. Local optimization on graphs, Discrete Aplied Mathematics 23, pp. 157 - 178, 1989. Erratum: 46, pp. 93-94, 1993.

[15] N. Megiddo, and C. Papadimitriou. On total functions, existence theorems, and computational complexity, Theoretical Computer Science 81, pp. 317324, 1991.

[16] M.Santha and M. Szegedy. Quantum and classical query complexities of local search are polynomially related, Proceedings of the thirty-sixth annual ACM symposium on Theory of computing, pp. 494-501, 2004.

[17] R. Spalek, M. Szegedy. All quantum adversary methods are equivalent. quant-ph/0409116.

[18] S. Zhang. On the power of Ambainis lower bounds, Proceedings of the 31st International Colloquium on Automata, Languages and Programming, pp. 1238-1250. Invited to Theoretical Computer Science.

Appendix

A Proof of Lemma 6

Recall that suppose the j-th ball is put into i_j-th bin, and $n_i \equiv 1$ means n_i is odd, $n_i \equiv 0$ means n_i is even.

Proof First, it is easy to see that for any $b_1, ..., b_m \in \{0, 1\}$ and any $i^* \in [m]$, it holds that

$$p_{i^*}^{(t)}[b_1, ..., b_m] \leq \frac{m}{m-1} p^{(t)}[b_1, ..., b_m] \tag{34}$$

Actually,

$$\begin{aligned} p^{(t)}[b_1, ..., b_m] &= \mathbf{Pr}[i_1 = i^*]\mathbf{Pr}[(i_1, ..., i_t) \text{ generates } (b_1, ..., b_m) | i_1 = i^*] && (35) \\ &\quad + \mathbf{Pr}[i_1 \neq i^*]\mathbf{Pr}[(i_1, ..., i_t) \text{ generates } (b_1, ..., b_m) | i_1 \neq i^*] && (36) \\ &= \frac{1}{m} p^{(t-1)}[b_1, ... b_{i^*-1}, 1 - b_{i^*}, b_{i^*+1}, ..., b_m] + \frac{m-1}{m} p_{i^*}^{(t)}[b_1, ..., b_m] && (37) \\ &\geq \frac{m-1}{m} p_{i^*}^{(t)}[b_1, ..., b_m] && (38) \end{aligned}$$

So to prove the lemma, it is enough to show the same upper bound for $p^{(t)}[b_1, ..., b_m]$.

We start with several simple observations. First, we assume that t and $\sum_{i=1}^m b_i$ have the same parity, because otherwise the probability is 0 and the lemma holds trivially. Second, by the symmetry, any permutation of $b_1, ..., b_m$ does not change $p^{(t)}[(b_1, ..., b_m)]$. Third, $p^{(t)}[(b_1, ..., b_m)]$ decreases if we replace two 1's in $b_1, ..., b_m$ by two 0's. Precisely, if we have two b_i's being 1, say $b_1 = b_2 = 1$, then $[(b_1, ..., b_m)] < p^{(t)}[(0, 0, b_3, ..., b_m)]$. In fact, note that

$$\begin{aligned} p^{(t)}[(b_1, ..., b_m)] &= \frac{1}{m^t} \sum_{\substack{n_1 + ... + n_m = t \\ n_i \equiv b_i, i \in [m]}} \frac{t!}{n_1! ... n_m!} && (39) \\ &= \frac{1}{m^t} \sum_{\substack{n_3 + ... + n_m \leq t \\ n_i \equiv b_i, i = 3, ..., m}} \left(\frac{t!}{(n_1 + n_2)! n_3! ... n_m!} \sum_{\substack{n_1 + n_2 = t - n_3 - ... - n_m \\ n_i \equiv b_i, i = 1, 2}} \frac{(n_1 + n_2)!}{n_1! n_2!} \right) && (40) \end{aligned}$$

where as usual, let $0! = 1$. If $n_3 + ... + n_m < t$, then

$$\sum_{\substack{n_1 + n_2 = t - n_3 - ... - n_m \\ n_i \equiv 1, i = 1, 2}} \frac{(n_1 + n_2)!}{n_1! n_2!} = \sum_{\substack{n_1 + n_2 = t - n_3 - ... - n_m \\ n_i \equiv 0, i = 1, 2}} \frac{(n_1 + n_2)!}{n_1! n_2!} \tag{41}$$

If $n_3 + ... + n_m = t$, then the only possible (n_1, n_2) is $(0, 0)$, so

$$\sum_{\substack{n_1 + n_2 = t - n_3 - ... - n_m \\ n_i \equiv 1, i = 1, 2}} \frac{(n_1 + n_2)!}{n_1! n_2!} = 0, \qquad \sum_{\substack{n_1 + n_2 = t - n_3 - ... - n_m \\ n_i \equiv 0, i = 1, 2}} \frac{(n_1 + n_2)!}{n_1! n_2!} = 1. \tag{42}$$

Thus $p^{(t)}[(1, 1, b_3, ..., b_m)] < p^{(t)}[(0, 0, b_3, ..., b_m)]$.

By the observations, it is sufficient to prove the lemma for the case $p^{(t)}[(0, ..., 0)]$ if t is even, and for the case $p^{(t)}[(1, 0, ..., 0)]$ if t is odd. Note that if t is even, then

$$p^{(t)}[(0, ..., 0)] = \sum_{i=1}^{m} \mathbf{Pr}[i_1 = i]\mathbf{Pr}[(i_2, ..., i_t) \text{ generates } (e_i)] \tag{43}$$

where e_i is the m-long vector with only coordinate i being 1 and all other coordinates being 0. By the symmetry, $p^{(t-1)}[e_1] = ... = p^{(t-1)}[e_m]$, thus $p^{(t)}[(0, ... 0)] = p^{(t-1)}[e_1] = p^{(t-1)}[1, 0, ..., 0]$. Therefore, it is enough to show the lemma for even t.

We now express $p^{(t)}[0,...,0]$ in two ways. One is to prove the first case ($t \leq 10$) in the lemma, and the other is for the second case ($10 < t \leq m^2$) and the third case ($t > m^2$) in the lemma.

To avoid confusion, we write the number m of bins explicitly as subscript: $p_m^{(t)}[b_1,...,b_m]$. We consider which bin(s) the first two balls is put into.

$$p_m^{(t)}[0,...,0] = \mathbf{Pr}[i_1 = i_2] p_m^{(t-2)}[0,...,0] + \mathbf{Pr}[i_1 \neq i_2] p_m^{(t-2)}[1,1,0,...,0] \tag{44}$$

$$= \frac{1}{m} p_m^{(t-2)}[0,...,0] + \frac{m-1}{m} p_m^{(t-2)}[1,1,0,...,0] \tag{45}$$

To compute $p_m^{(t-2)}[1,1,0,...,0]$, we consider to put $(t-2)$ balls in m bins. By the analysis of the third observations above, we know that

$$p_m^{(t-2)}[0,...,0] - p_m^{(t-2)}[1,1,0,...,0] \tag{46}$$

$$= \mathbf{Pr}[n_1 = n_2 = 0, n_3 \equiv 0, ..., n_m \equiv 0] \tag{47}$$

$$= \mathbf{Pr}[n_1 = n_2 = 0] \mathbf{Pr}[n_3 \equiv 0, ..., n_m \equiv 0 | n_1 = n_2 = 0] \tag{48}$$

$$= \left(\frac{m-2}{m}\right)^{t-2} p_{m-2}^{(t-2)}[0,...,0] \tag{49}$$

Therefore,

$$p_m^{(t)}[0,...,0] = p_m^{(t-2)}[0,...,0] - \frac{m-1}{m} \left(\frac{m-2}{m}\right)^{t-2} p_{m-2}^{(t-2)}[0,...,0] \tag{50}$$

Now using the above recursive formula and the base case $p_m^{(2)}[0,...,0] = 1/m$, it is easy (but tedious) to prove by calculations that $p_m^{(t)}[0,...,0] = ((t-1)!!/m^{\frac{t}{2}})(1-o(1))$ for even $t \leq 10$. This proves the first case in the lemma.

For the rest two cases, consider the generating function $(x_1+...+x_m)^t = \sum_{n_1+...+n_m=t} \binom{t}{n_1,...,n_m} x_1^{n_1}...x_m^{n_m}$. If $x_i \in \{-1,1\}$, then $(x_1+...+x_m)^t = \sum_{n_1+...+n_m=t} \binom{t}{n_1,...,n_m} (-1)^{|\{i: x_i=-1, n_i \equiv 1\}|}$. We sum it over all $x_1...x_n \in \{-1,1\}^n$. Note that for those $(n_1,...,n_m)$ that has some $n_{i_0} \equiv 1$, it holds due to the cancelation that $\sum_{x_1,...,x_m \in \{-1,1\}} (-1)^{|\{i: x_i=-1, n_i \equiv 1\}|} = 0$. On the other hand, if all n_i's are even, then $\sum_{x_1,...,x_m \in \{-1,1\}} (-1)^{|\{i: x_i=-1, n_i \equiv 1\}|} = 2^m$. Thus we have $\sum_{x_1,...,x_m \in \{-1,1\}} (x_1+...+x_m)^t = 2^m \sum_{\substack{n_1+...+n_m=t \\ n_i \equiv 0, i \in [m]}} \binom{t}{n_1,...,n_m}$. Therefore

$$p^{(t)}[0,...,0] = \frac{1}{m^t} \sum_{\substack{n_1+...+n_m=t \\ n_i \equiv 0, i \in [m]}} \binom{t}{n_1,...,n_m} \tag{51}$$

$$= \frac{1}{2^m m^t} \sum_{x_1,...,x_m \in \{-1,1\}} (x_1+...+x_m)^t \tag{52}$$

$$= \frac{1}{2^m m^t} \sum_{i=0}^{m} \binom{m}{i} (m-2i)^t = \frac{1}{2^m} \sum_{i=0}^{m} \binom{m}{i} \left(1 - \frac{2i}{m}\right)^t. \tag{53}$$

It follows that $p^{(t)}[0,...,0]$ decreases with t, and this proves the second case of the lemma with the help of the first case. And if $t > m^2/2$, then

$$p^{(t)}[0,...,0] \leq \frac{1}{2^m} \left(2 + \left(1 - \frac{2}{m}\right)^t \sum_{i=1}^{m-1} \binom{m}{i}\right) < 2/2^m + e^{-m} = O(1/2^m) \tag{54}$$

This proves the third case of the lemma. □

B Proof of Proposition 7

Proof We consider two settings. One is as in the definition of the short walk, where we have only n points $0,...,n-1$, and points 0 and $n-1$ are two barriers[3]. Another is the same except that the

[3] Here we let the n points be $0,...,n-1$ instead of $1,...,n$ just to make the later calculation cleaner

barriers are removed, and we have infinite points in a line. For each t-bit binary string $x = x_1...x_t$, we use P_i^x and Q_i^x to denote the two paths that starting at i and walk according to x in the two settings. Precisely, at step s, Q_i^x goes left if $x_s = 0$ and goes right if $x_s = 1$. P_i^x goes in the same way except that it will stand still if the point is currently at left (or right) end and it still wants to go left (or right). If the end point of P_i^x is j, then we write $i \to_t^{P,x} j$. Let $X_{ij}^{(t),P}$ be the set of $x \in \{0,1\}^t$ *s.t.* $i \to_t^{P,x} j$, and put $n_{ij}^{(t),P} = |X_{ij}^{(t),P}|$. Then by definition, $p_{ij}^{(t)} = n_{ij}^{(t),P}/2^t$. The notations $i \to_t^{Q,x} j$, $X_{ij}^{(t),Q}$ and $n_{ij}^{(t),Q}$ are similarly defined, with the corresponding P changed to Q. Note that $n_{ij}^{(t),Q} = \binom{t}{t/2+(j-i)/2}$ if $j-i$ and t have the same parity, and 0 otherwise. We now want to upper bound $n_{ij}^{(t),P}$ in terms of $n_{ij}^{(t),Q}$.

For a path P_i^x, if at some step it is at point 0 and wants to go left, we say it *attempts to pass the left barrier*. Similarly for the right barrier. We say a path is in the $\{a_s, b_s\}_{s=1}^l$ category if it first attempts to pass the left barrier for a_1 times, and then attempts to pass the right barrier for b_1 times, and so on. We call each round a stage s, which begins at the time that P_i^x attempts to pass the left barrier for the $(a_1 + ... + a_{s-1} + 1)$-th time, and ends right before the time that P_i^x attempts to pass the left barrier for the $(a_1 + ... + a_s + 1)$-th time. We also split each stage s into two halves, cutting at the time right before the path attempts to pass the right barrier for the $(b_1 + ... + b_{s-1} + 1)$-th time. Note that a_1 may be 0, which means that the path first attempts to pass the right barrier. Also b_l may be 0, which means the the last barrier the path attempts to pass is the left one. But all other a_i, b_i's are positive. Also note that in the case of $l = 0$, the path never attempts to pass either barrier. We partition $X_{ij}^{(t),P}$ as

$$X_{ij}^{(t),P} = \bigcup_{l,\ \{a_s,b_s\}_{s=1}^l} X_{ij}^{(t),P}[\{a_s, b_s\}_{s=1}^l] \tag{55}$$

where $X_{ij}^{(t),P}[\{a_s, b_s\}_{s=1}^l]$ contains those paths in the category $\{a_s, b_s\}_{s=1}^l$. Put $n_{ij}^{(t),P}[\{a_s, b_s\}_{s=1}^l] = |X_{ij}^{(t),P}[\{a_s, b_s\}_{s=1}^l]|$, thus $n_{ij}^{(t),P} = \sum_l \sum_{\{a_s,b_s\}_{s=1}^l} n_{ij}^{(t),P}[\{a_s, b_s\}_{s=1}^l]$.

Now consider the corresponding paths in $X_{ij}^{(t),Q}$. The following observation relates P_i^x and Q_i^x.

Observation 1 *For each $x \in X_{ij}^{(t),P}[\{a_s, b_s\}_{s=1}^l]$, the following two properties hold.*

1. *In the first half of stage s, the path Q_i^x touches (from right) but does not cross the point $\alpha_s = \sum_{r=1}^{s-1}(b_r - a_r) - a_s$.*
2. *In the second half of stage s, the path Q_i^x touches (from left) but does not cross the point $\beta_s = n - 1 + \sum_{r=1}^{s}(b_r - a_r)$*
3. *The path Q_i^x ends at $\gamma = j + \sum_{s=1}^{l}(b_s - a_s)$*

We let $Y_{i\gamma}^{(t),Q}[\{\alpha_s, \beta_s\}_{s=1}^l]$ contain those $x \in \{0,1\}^t$ satisfying the three conditions in the above observation, and denote by $m_{i\gamma}^{(t),Q}[\{\alpha_s, \beta_s\}_{s=1}^l]$ the size of the set $Y_{i\gamma}^{(t),Q}[\{\alpha_s, \beta_s\}_{s=1}^l]$. Thus the observation says $X_{ij}^{(t),P}[\{\alpha_s, \beta_s\}_{s=1}^l] \subseteq Y_{ij}^{(t),Q}[\{\alpha_s, \beta_s\}_{s=1}^l]$, and therefore we have $n_{ij}^{(t),P}[\{a_s, b_s\}_{s=1}^l] \leq m_{i\gamma}^{(t),Q}[\{\alpha_s, \beta_s\}_{s=1}^l]$. Now for each $x \in Y_{i\gamma}^{(t),Q}[\{\alpha_s, \beta_s\}_{s=1}^l]$, if we change the condition 1 in the case $s = 1$ by allowing the path to cross the point α_1, and let $Z_{i\gamma}^{(t),Q}[\{\alpha_s, \beta_s\}_{s=1}^l]$ be the new set satisfying the new conditions, then $m_{i\gamma}^{(t),Q}[\{\alpha_s, \beta_s\}_{s=1}^l] = |Z_{i\gamma}^{(t),Q}[\{\alpha_s, \beta_s\}_{s=1}^l]| - |Z_{i\gamma}^{(t),Q}[\alpha_1 - 1, \beta_1, \{\alpha_s, \beta_s\}_{s=2}^l]|$. In other words, the set of paths touches (from right) but does not cross α_1 is the set of paths touches or crosses α_1 minus the set of paths touches or crosses $\alpha_1 - 1$.

Now we calculate $|Z_{i\gamma}^{(t),Q}[\{\alpha_s, \beta_s\}_{s=1}^l]|$ by the so-called reflection rule. Suppose the first time that Q_i^x touches α_1 is t_1. We reflect the first t_1 part of the path Q_i^x with respect to the point α_1. Precisely, let $y = (1 - x_1)...(1 - x_{t_1})x_{t_1+1}...x_t$, then the paths Q_i^x and $Q_{2\alpha_1 - i}^y$ merge at time t_1. And it is easy to check that it is a 1-1 correspondence between $Z_{i\gamma}^{(t),Q}[\{\alpha_s, \beta_s\}_{s=1}^l]$ and $Y_{2\alpha_1 - i,\gamma}^{(t),Q}[\beta_1, \{\alpha_s, \beta_s\}_{s=2}^l]$, Here $Y_{2\alpha_1 - i,\gamma}^{(t),Q}[\beta_1, \{\alpha_s, \beta_s\}_{s=2}^l]$ is the set of paths starting at $2\alpha_1 - i$, satisfying (a) the condition 2 at the first stage, (b) both conditions 1 and 2 at the rest $l-1$ stages, and (c) condition 3. So

$$|Z_{i\gamma}^{(t),Q}[\{\alpha_s,\beta_s\}_{s=1}^l]| = |Y_{2\alpha_1-i,\gamma}^{(t),Q}[\beta_1,\{\alpha_s,\beta_s\}_{s=2}^l]| = m_{2\alpha_1-i,\gamma}^{(t),Q}[\beta_1,\{\alpha_s,\beta_s\}_{s=2}^l] \quad (56)$$

$$= m_{-2a_1-i,\gamma}^{(t),Q}[\beta_1,\{\alpha_s,\beta_s\}_{s=2}^l] \quad (57)$$

$$= m_{-a_1-i,\gamma+a_1}^{(t),Q}[\beta_1+a_1,\{\alpha_s+a_1,\beta_s+a_1\}_{s=2}^l] \quad (58)$$

where (57) is due to the fact that $\alpha_1=-a_1$, and (58) is because that the number of the paths does not change if we move all the paths right by a_1. Similarly, we have

$$|Z_{i\gamma}^{(t),Q}[\alpha_1-1,\beta_1,\{\alpha_s,\beta_s\}_{s=2}^l]| = m_{2\alpha_1-2-i,\gamma}^{(t),Q}[\beta_1,\{\alpha_s,\beta_s\}_{s=2}^l] \quad (59)$$

$$= m_{-a_1-2-i,\gamma+a_1}^{(t),Q}[\beta_1+a_1,\{\alpha_s+a_1,\beta_s+a_1\}_{s=2}^l] \quad (60)$$

Therefore,

$$n_{ij}^{(t),P}[\{a_s,b_s\}_{s=1}^l] \le m_{i\gamma}^{(t),Q}[\{\alpha_s,\beta_s\}_{s=1}^l] \quad (61)$$

$$= m_{-2a_1-i,\gamma}^{(t),Q}[\beta_1,\{\alpha_s,\beta_s\}_{s=2}^l] - m_{-2a_1-2-i,\gamma}^{(t),Q}[\beta_1,\{\alpha_s,\beta_s\}_{s=2}^l] \quad (62)$$

$$= m_{-a_1-i,\gamma+a_1}^{(t),Q}[\beta_1+a_1,\{\alpha_s+a_1,\beta_s+a_1\}_{s=2}^l] \quad (63)$$

$$- m_{-a_1-2-i,\gamma+a_1}^{(t),Q}[\beta_1+a_1,\{\alpha_s+a_1,\beta_s+a_1\}_{s=2}^l] \quad (64)$$

Now for any fixed $l>1$, we consider those categories with $a_1>0$ and $b_l>0$. Other cases can handled similarly. Note that $\alpha_s+a_1 = b_1+\sum_{r=2}^{s-1}(b_r-a_r)-a_s$, $\beta_s+a_1 = n-1+\sum_{r=2}^{s}(b_r-a_r)$ and $\gamma+a_1 = j+b_1+\sum_{r=2}^{s}(b_r-a_r)$ are all functions of $(b_1,a_2,b_2,...,a_l,b_l)$, not of a_1 any more. Therefore,

$$\sum_{a_1,b_1,...,a_l,b_l>0} n_{ij}^{(t),P}[\{a_s,b_s\}_{s=1}^l] \quad (65)$$

$$\le \sum_{b_1,...,a_l,b_l>0}\sum_{a_1>0}(m_{-a_1-i,\gamma+a_1}^{(t),Q}[\beta_1+a_1,\{\alpha_s+a_1,\beta_s+a_1\}_{s=2}^l] \quad (66)$$

$$- m_{-a_1-2-i,\gamma+a_1}^{(t),Q}[\beta_1+a_1,\{\alpha_s+a_1,\beta_s+a_1\}_{s=2}^l]) \quad (67)$$

$$= \sum_{b_1,...,a_l,b_l>0}(m_{-1-i,\gamma+a_1}^{(t),Q}[\beta_1+a_1,\{\alpha_s+a_1,\beta_s+a_1\}_{s=2}^l] \quad (68)$$

$$+ m_{-2-i,\gamma+a_1}^{(t),Q}[\beta_1+a_1,\{\alpha_s+a_1,\beta_s+a_1\}_{s=2}^l]) \quad (69)$$

Note that due to the parity, only one of $m_{-1-i,\gamma+a_1}^{(t),Q}[\beta_1+a_1,\{\alpha_s+a_1,\beta_s+a_1\}_{s=2}^l]$ and $m_{-2-i,\gamma+a_1}^{(t),Q}[\beta_1+a_1,\{\alpha_s+a_1,\beta_s+a_1\}_{s=2}^l]$ is nonzero. So the summation of them two items is equal to the maximum of them. Now using the similar methods, *i.e.* reflecting with respect to points $(n-1+b_1)$ and $(n+b_1)$, moving the paths left by b_1, and finally collapsing the telescope, we can get

$$\sum_{b_1,...,a_l,b_l>0} m_{-1-i,\gamma+a_1}^{(t),Q}[\beta_1+a_1,\{\alpha_s+a_1,\beta_s+a_1\}_{s=2}^l] \quad (70)$$

$$= \sum_{a_2,b_2,...,a_l,b_l>0}\max\{m_{2n+i,\gamma+a_1-b_1}^{(t),Q}[\{\alpha_s+a_1-b_1,\beta_s+a_1-b_1\}_{s=2}^l], \quad (71)$$

$$m_{2n+i+1,\gamma+a_1-b_1}^{(t),Q}[\{\alpha_s+a_1-b_1,\beta_s+a_1-b_1\}_{s=2}^l]\} \quad (72)$$

and

$$\sum_{b_1,...,a_l,b_l>0} m_{-2-i,\gamma+a_1}^{(t),Q}[\beta_1+a_1,\{\alpha_s+a_1,\beta_s+a_1\}_{s=2}^l] \quad (73)$$

$$= \sum_{a_2,b_2,...,a_l,b_l>0}\max\{m_{2n+i+2,\gamma+a_1-b_1}^{(t),Q}[\{\alpha_s+a_1-b_1,\beta_s+a_1-b_1\}_{s=2}^l], \quad (74)$$

$$m_{2n+i+3,\gamma+a_1-b_1}^{(t),Q}[\{\alpha_s+a_1-b_1,\beta_s+a_1-b_1\}_{s=2}^l]\} \quad (75)$$

We continue this process, and finally it is

$$\sum_{a_1,b_1,...,a_l,b_l>0} n_{ij}^{(t),P}[\{a_s,b_s\}_{s=1}^l] \le \max\{n^{(t),Q}_{2ln+i+h,\gamma+\sum_{s=1}^l(a_s-b_s)} : h=0,...,4l-1\} \tag{76}$$

$$= \max\{n^{(t),Q}_{2ln+i+h,j} : h=0,...,4l-1\} \tag{77}$$

$$= n^{(t),Q}_{2ln+i,j} \tag{78}$$

$$\le \binom{t}{\frac{t}{2}+\frac{j-i-2ln}{2}} \tag{79}$$

Thus

$$\sum_{l>0}\sum_{a_1,b_1,...,a_l,b_l>0} n_{ij}^{(t),P}[\{a_s,b_s\}_{s=1}^l] \le \sum_{l>0}\binom{t}{\frac{t}{2}+\frac{j-i}{2}-ln} = \begin{cases} O(\frac{2^t}{\sqrt{t}}) & \text{if } t<n^2 \\ O(\frac{\sqrt{t}}{n}\frac{2^t}{\sqrt{t}}) = O(\frac{2^t}{n}) & \text{if } t\ge n^2 \end{cases} \tag{80}$$

For other categories that $a_1=0$ or $b_l=0$, the same result can be proved similarly, and the $l=0$ is easy since $n_{ij}^{(t),Q} = O(2^t/\sqrt{t})$. Putting all things together, we get the result

$$p_{ij}^{(t)} = \begin{cases} O(1/\sqrt{t}) & \text{if} \quad t\le n^2 \\ O(1/n) & \text{if} \quad t>n^2 \end{cases} \tag{81}$$

for any i and j, which completes our proof. □

C Further explanations of the construction in Section 4.3

In this section, we further explain the construction in Section 4.3. In particular, we shall make the claim more clear that we can think of the construction the same as a long grid $[\alpha]^{d-1}\times[(n'-2\alpha)\beta^{d-1}]$. Actually, what we care about is, as before, the probability that the random walk starting from a point $x=x_0...x_{d-1}$ passes another point $x'=x_0...x_{d-1}$ after exactly $t'-t$ steps. Here t is the time that the random path passes x and t' is the time that the path passes x'. Note that t is fixed and known by x itself; similarly for t'. Denote this probability by $\mathbf{Pr}[x\to x']$. Suppose $x_i=(k_i-1)\alpha+y_i$ and $x'_i=(k'_i-1)\alpha+y'_i$ for $i\in\{0,...,d-2\}$.

We first consider the case that one of the two points, say x' is on a block-changing segment. Since different block-changing segments never intersect, a path passes x' if and only if the path passes the boundary point x'' at the beginning of the block-changing segment that x' is in. Also note that the time that the path passes x'' is also t' because the time does not elapse on the block-changing segment. So it holds that $\mathbf{Pr}[x\to x']=\mathbf{Pr}[x\to x'']$, and it is enough to consider the case that both x and x' are not in clock-changing segments.

Now suppose both x and x' are not in clock-changing segments. In general, x and x' may be not in the same block , so going from x to x' needs to change blocks. Recall that to change from the block $(k_0,...,k_{d-2})$ to the next one, only one k_i changes by increasing or decreasing by 1. Suppose that to go to x' from x, we change blocks for c times, by changing $k_{i_1},k_{i_2},...,k_{i_c}$ in turn. Let $n_j=|\{s\in[c]:i_s=j\}|$. Note that to get to x' from x after $t'-t$ steps, the coordinate j needs to be x'_j after $t'-t$ steps for each coordinate $j\in\{0,...,d-2\}$. It is not hard to see that if a block-changing needs to change k_j, then only the coordinate j gets reflected within the current block. That is, suppose the coordinate j is $(k_j-1)\alpha+y_j$ before the block-changing, then it changes to $(k_j-1)\alpha+\alpha+1-y_j$ after the block-changing. So if $c=1$, then $\mathbf{Pr}[x\to x']$ is equal to the probability that a random walk in $[\alpha]^{d-1}$ starting from $y_0...y_{d-2}$ stops at $y''_0...y''_{d-2}$ after $t'-t$ steps, where $y''_j=y'_j$ if $j\ne i_1$ and $y''_{i_1}=(k_{i_1}-1)\alpha+\alpha+1-y'_{i_1}$. For general c, $\mathbf{Pr}[x\to x']$ is equal to the probability that a random walk in $[\alpha]^{d-1}$ starting from $y_0...y_{d-2}$ stops at $y''_0...y''_{d-2}$ after $t'-t$ steps, where $y''_j=y'_j$ if n_j is even and $y''_j=(k_j-1)\alpha+\alpha+1-y'_j$ if n_j is odd. Note that the latter probability has nothing to do with the block-changing; it is just the same as we have a clock space $[(n'-2\alpha)\beta^{d-1}]$ to record the random walk on $[\alpha]^{d-1}$. Thus we can use Proposition 7 to upper bound this probability and further the proof of the lower bound.

ECCC ISSN 1433-8092
http://www.eccc.uni-trier.de/eccc
ftp://ftp.eccc.uni-trier.de/pub/eccc
ftpmail@ftp.eccc.uni-trier.de, subject 'help eccc'