Time Lower Bounds for Distributed Distance Oracles*

Taisuke Izumi[1] and Roger Wattenhofer[2]

[1] Nagoya Institute of Technology,
Gokiso-cho, Showa-ku, Nagoya, Aichi, 466-8555, Japan
t-izumi@nitech.ac.jp
[2] ETH Zurich, 8092 Zurich, Switzerland
wattenhofer@ethz.ch

Abstract. *Distributed distance oracles* consist of a labeling scheme which assigns a label to each node and a local data structure deployed to each node. When a node v wants to know the distance to a node u, it queries its local data structure with the label of u. The data structure returns an estimated distance to u, which must be larger than the actual distance but can be overestimated. The accuracy of the distance oracle is measured by *stretch*, which is defined as the maximum ratio between actual distances and estimated distances over all pairs (u, v).

In this paper, we focus on the time complexity of constructing distributed distance oracles with a given stretch. We show a number of time lower bounds depending on the stretch:

- Under the assumption that the popular combinatorial girth conjecture is true, any distributed algorithm constructing oracles with stretch $2t$ requires $\tilde{\Omega}(n^{1/(t+1)})$ rounds in unweighted graphs. This bound holds even if we only consider constant diameter graphs.
- For oracles with stretch $2t$ in weighted graphs, we have a lower bound of $\Omega(n^{\frac{1}{2}+\frac{1}{5t}})$ rounds, assuming the girth conjecture. This bound holds even if we only consider $O(\log n)$ diameter graphs.
- If we restrict the label size of oracles to $o(n^{\epsilon})$ bits, where $\epsilon = 1/2t(t+1)$ in unweighted graphs and $\epsilon = (1/5t^2)$ in weighted graphs, the same lower bounds are obtained without assuming the girth conjecture.

To the best of our knowledge, this paper is the first that exhibits a non-trivial trade-off between time and stretch for distributed distance oracles.

1 Introduction

1.1 Background

The primary objective of routing protocols is to identify paths from sources to destinations, in order to route packets efficiently. While there are a number of

* This work is supported in part by KAKENHI No.25106507 and No.25289114.

M.K. Aguilera et al. (Eds.): OPODIS 2014, LNCS 8878, pp. 60–75, 2014.

© Springer International Publishing Switzerland 2014

criteria to measure efficiency (delay, bandwidth, reliability, and so on), the most popular choice for selecting a good path is the path's length (i.e., distance between two nodes). In other words, Internet routing is still often synonymous to shortest path routing. A well-known example is the distance-vector routing protocol BGP. Often, real-world routing protocols weigh the edges of the network to measure distance more precisely. Sometimes, this weight information is somewhat hidden. In the case of BGP, for example, a technique called AS-path prepending is used, where a node includes itself in the route several times in order to give an edge more weight, i.e. to discourage other nodes from routing through it.

Regarding distributed complexity, many distance problems are recognized as so-called *global* problems. That is, their distributed time-complexity is $\Omega(\Delta)$, where Δ is the unweighted (hop-count) diameter of the network. A common naive solution for such global problems is the centralized approach: A single node aggregates the whole topological information of the network (and the weights of all edges in the case of weighted graphs), and computes the solution locally. This solution gives a $O(\Delta)$-time matching upper bound for the model with unbounded communication on each edge. However, for large networks, this unbounded (often also called LOCAL) message passing model becomes unreasonable. Instead one should assume that communication messages are limited. An established model for distributed computation is the so-called CONGEST message passing model. It allows each message to have at most $O(\log n)$ bits, where n is the number of all nodes in the network. In this model, it is known that the conventional all-pairs distance computation or approximation requires $\tilde{\Omega}(n)$ rounds [5,11] even in unweighted graphs with constant diameters.[1] On the other hand, near tight upper bounds are also known for both weighted and unweighted graphs. In unweighted graphs, there is an algorithm constructing all-pair shortest paths in $O(n)$ rounds [9,6], and in weighted graphs, there is an algorithm computing an $(1 + o(1))$-approximation of all-pairs distances in $\tilde{O}(n)$ rounds [11]. These results imply that all-pairs distance computation is an expensive task, even in the approximation case.

1.2 Distance Oracles

The inherent difficulty behind the all-pairs distance computation is that each node must fill out its own distance table of $n-1$ entries (one of which corresponds to the distance to some other node). In other words, it is inherently necessary that each node must receive $\Omega(n)$ bits of information to fill out the table of size $\Theta(n)$. However, if we can have a more compact representation of distance tables, its construction can be achieved in sublinear time. This observation yields to the problem of *distributed distance oracles*. A *distance oracle* is a subquadratic-size data structure storing all-pairs approximated distances, which was originally introduced in the context of centralized algorithms [14]. A distributed distance

[1] The tilde complexity notation $\tilde{O}(f(n))$ hides a polylogarithmic factor in n, usually, in this line of work, caused by the $O(\log n)$ bits allowed in each message.

oracle consists of a labeling scheme giving a label to each node and a local data structure deployed to each node v in the network. When a node v wants to know the distance to another node u, v queries its local data structure with the label of u. The data structure returns an estimated distance to u, which must be larger than the actual distance but can be overestimated. The approximation factor of distance oracles is also called *stretch*, which is defined as the maximum ratio between actual distances and estimated distances over all pairs (u, v).

There have been two results about the construction time of distributed distance oracles so far. The first one is by Das Sarma et al. [1], which gives an algorithm guaranteeing stretch $2t-1$, $\tilde{O}(n^{1/t})$-bit label size, and $\tilde{O}(n^{1/t}\Delta')$-round construction time, where t is a parameter trading time, space, and stretch, and Δ' is the shortest-path diameter of the graph (i.e., the maximum hop length over all-pairs shortest paths.[2]. Since Δ' can become linear of n at the worst-case, the construction time of this algorithm can be superlinear. A second paper by Lenzen and Patt-Shamir [8] proposes an algorithm with stretch $2t(8t-3)$, $O(t(\log n))$-bit label size, and $\tilde{O}(n^{1/2+1/2k} + \Delta)$-round construction time. It also shows that any distributed distance oracle algorithm achieving an arbitrary non-trivial stretch must have $\tilde{\Omega}(\sqrt{n})$ construction time.

1.3 Our Contribution

In this paper, we present several time lower bounds for the construction of distributed distance oracles. The primary results of our paper are new improved lower bounds depending on stretch. More precisely, our contributions are as follows:

- Under the assumption that the popular combinatorial girth conjecture is true, any distributed algorithm constructing oracles with stretch $2t$ requires $\tilde{\Omega}(n^{1/(t+1)})$ rounds in unweighted graphs. This bound holds even if we only consider constant diameter graphs.
- For oracles with stretch $2t$ in weighted graphs, we have a lower bound of $\Omega(n^{\frac{1}{2}+\frac{1}{5t}})$ rounds, assuming the girth conjecture. This bound holds even if we only consider $O(\log n)$ diameter graphs.
- If we restrict the label size of oracles to $o(n^{\epsilon})$ bits, where $\epsilon = 1/2t(t+1)$ in unweighted graphs and $\epsilon = (1/5t^2)$ in weighted graphs, the same lower bounds are obtained without assuming the girth conjecture.

To the best of our knowledge, these are the first results that exhibit a non-trivial trade-off between construction time and stretch.

1.4 Related Work

For unweighted graphs, there has been a lot of progress to understand the distributed complexity [9,5,6,12,2] of distance problems such as the single-source

[2] Note that $\Delta \leq \Delta'$ always holds because for the pair (u, v) giving the hop-count diameter path, the hop-length of the shortest path between u and v cannot be shorter than Δ.

shortest paths, all-pairs shortest paths, diameter, and distance oracles. A first paper by Frischknecht et al. [5] showed an $\tilde{\Omega}(n)$-time lower bound for the exact diameter computation in unweighted networks with constant diameters. A matching upper bound was shown by Holzer at al. [6], and Lenzen and Peleg [9]; they concurrently and independently proposed almost the same $O(n)$-time algorithm for all-pairs shortest paths. The hardness of the approximated diameter computation is also considered. An easy solution for a 2-approximation of the diameter is to construct a shortest path tree rooted at an arbitrary node u. Since shortest path trees and breadth-first search (BFS) are equivalent in unweighted networks, a 2-approximation is trivially achieved in $O(\Delta)$ time by running a simple BFS-tree construction. A result by [5] also showed that any 3/2-approximation algorithm for the diameter problem requires $\tilde{\Omega}(\sqrt{n})$ time. This lower bound is improved to $\tilde{\Omega}(n)$ by [6]. Interestingly, regarding upper bounds, Peleg et al. showed that a 3/2-approximated value of Δ is computable in $\tilde{O}(\sqrt{n}\Delta)$ time [12]. A recent paper [9] improves this time bound for a 3/2-approximation to an additive $\tilde{O}(\sqrt{n} + \Delta)$ time. Holzer and et al. [6] show a more accurate approximation algorithm for the network diameter problem with $O(n/\Delta + \Delta)$ running time. Its approximation factor is $(1 + \epsilon)$ for an arbitrary small constant $\epsilon < 1$.

While a rich literature exists for unweighted networks, only a few papers consider distance problems in weighted networks. To the best of our knowledge, there are three papers directly related to weighted graphs. Das Sarma et al. [1] and Lenzen and Patt-Shamir [8] we already discussed in the introduction. The paper by Lenzen and Patt-Shamir [8] also considers several related problems, including (all-pairs) shortest paths or diameter. In addition there is a very recent result by Nanongkai [11]. It proposes faster distributed approximation algorithms for single-source shortest paths and all-pairs shortest paths.

1.5 Roadmap

The paper is organized as follows: We introduce fundamental definitions and notations in Section 2. In Section 3, we give our lower bound proof for unweighted graphs. It is extended to the weighted case in Section 4. The case for bounded label size oracles is considered in Section 5. Finally in Section 6, we conclude this paper.

2 Preliminaries

2.1 Round-Based Synchronous Systems

A distributed system consists of n nodes interconnected with communication links. We model it by a weighted undirected graph $G = (V, E, w)$, where $V = \{v_0, v_1, \cdots, v_{n-1}\}$ is the set of nodes, $E \subseteq V \times V$ is the set of links (edges), and $w : E \to \mathbb{N}$ is the edge-weight function. Since we consider undirected graphs, $w(u, v) = w(v, u)$ holds for any $u, v \in V$. We also consider the system modeled by unweighted graphs, which is a special case of weighted graphs where every edge has weight one.

Executions of the system proceed with a sequence of consecutive rounds. In each round, each process sends a (possibly different) message to each neighbor, and within the round, all messages are received. After receiving its messages, each process performs local computation. Throughout this paper, we restrict the number of bits transmittable through any communication link per one round to $O(\log n)$ bits. This is known as the CONGEST model. Note that in weighted networks the weight of each edge does not imply the delay of communication. It is guaranteed that messages transferred through weighted edges reach their destinations within one round.

A path P between u and v is a sequence $u = u_0, u_1, \cdots u_k = v$ such that $(u_{i-1}, u_i) \in E$ holds for any i $(1 \leq i \leq k)$. The *distance* between u and v in graph G is the weighted length of the shortest path between them, which is denoted by $d_G(u, v)$.

2.2 Problem Definition

The *distributed distance oracle* is defined as the problem of constructing a labeling scheme $\lambda : V \to L$, where L is the domain of labels, and a local data structure $dest_v : L \to \mathbb{Z}$ deployed to each node $v \in V$, which locally computes the distance estimation from v by giving the label $\lambda(u)$ of any target node u. The value $dest_v(\lambda(u))$ returned by the local oracle at node v is always lower at least the actual distance $d_G(u, v)$. The *stretch* of a distributed distance oracle is defined as $\max_{u,v \in V} dest_v(\lambda(u))/d_G(v, u)$. The *label size* of a distributed distance oracle is defined as $\lceil \log |L| \rceil$.

3 Lower Bound for Unweighted Graphs

3.1 Two-Party Communication Complexity

Communication complexity, which was first introduced by Yao [15], reveals the amount of communication to compute a global function whose inputs are distributed in the network. The most successful scenario in communication complexity is *two-party* communication complexity, where two players, called Alice and Bob, have x-dimensional 0-1 vectors $\mathbf{a}$ and $\mathbf{b}$ respectively, and compute a global function $f : \{0, 1\}^x \times \{0, 1\}^x \to \{0, 1\}$. The communication complexity of a two-party protocol is the number of one-bit messages exchanged by the protocol for the worst case input (if the protocol is randomized, it is defined as the expected number of bits exchanged for the worst-case input). One of the most useful problems in communication complexity theory is *set-disjointness*:

Definition 1. *The x-bit set-disjointness function $disj_x : \{0, 1\}^x \times \{0, 1\}^x \to \{0, 1\}$ is defined as follows:*

$$disj_x(\mathbf{a}, \mathbf{b}) = \begin{cases} 1 \text{ if } \exists i \in [0, x-1] : a_i = b_i = 1, \\ 0 \text{ otherwise} \end{cases}$$

For this problem, the following theorem is known [13,7].

Theorem 1. *The communication complexity of the x-bit set-disjointness problem is $\Omega(x)$.*

In the following argument, we use a slightly different form of the set-disjointness problem: We first introduce a *base graph* $H = (W, F)$ such that $|W| = N$, $|F| = M$ for some value $N > 0$ and $M > 0$. Alice and Bob respectively have subsets F_a and F_b of F as their inputs. The goal of the two-party computation is to decide if $(W, F_a \cup F_b) = H$ holds or not. This problem is equivalent to the M-bit set-disjointness problem, i.e., each edge in G is one-bit entry of the set-disjointness, and $e \in F_a$ (resp. $e \in F_b$) implies that Alice's (resp. Bob's) corresponding bit is set to zero. Thus by Theorem 1, the communication complexity of this problem is $\Omega(M)$. In what follows, we refer to this form of the set-disjointness problem as the *graphic set-disjointness* over H. If an instance (F_a, F_b) satisfies $F_a \cup F_b = F$, we say that (F_a, F_b) is *disjoint*. Otherwise we say that (F_a, F_b) is *intersecting*. Two examples of the graphic set-disjointness are shown in Figure 1, where one instance is disjoint and another is intersecting. The black (resp. gray) lines represent the edges Alice and Bob have (resp. does not have), and the dotted line in the intersecting case is the the edge commonly lost by both players).

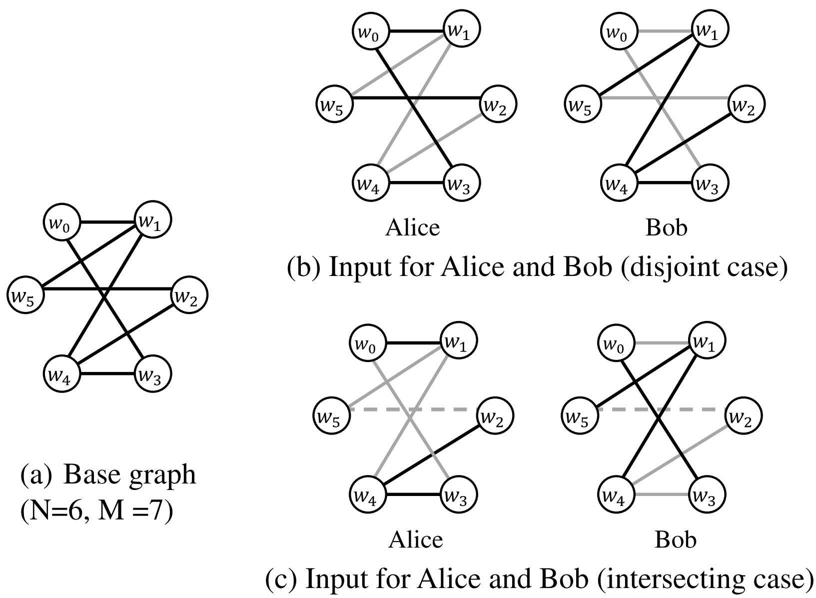

Fig. 1. Two examples of graphic set-disjointness instances

3.2 Gadget Construction

The core of the lower bound proof is a reduction from the graphic set-disjointness over some large-girth graph. The reduction scheme itself is similar to one introduced by [5]. This subsection shows the construction of the gadget for the reduction from the graphic set-disjointness over $H = (W, F)$. The constructed graph is denoted by $\Gamma_{H,\gamma}(F_a, F_b)$, where γ is a design parameter and (F_a, F_b) is any instance of the graphic set-disjointness over H. Letting $\Gamma_{H,\gamma}(F_a, F_b) = (V, E)$, V and E are constructed by the following steps:

1. The set of nodes V consists of two groups of N nodes $W^a = \{w^a_0, w^a_1, \cdots, w^a_{N-1}\}$ and $W^b = \{w^b_0, w^b_1, \cdots, w^b_{N-1}\}$.
2. For any i, $0 \leq i \leq N-1$, each pair (w^a_i, w^b_i) is connected by an edge. The path (of length one) added in this step is called an *intra-cluster* path.
3. Each pair $(w^a_i, w^a_j) \in W^a$ (resp. $(w^b_i, w^b_j) \in W^b$) is connected by a path of length γ if and only if $(w_i, w_j) \in F_a$ (resp. $(w_i, w_j) \in F_b$) $(\gamma > 0)$. The path added in this step is called an *inter-cluster* path.

Informally, $\Gamma_{H,\gamma}(F_a, F_b)$ behaves as the weighted version of graph $H' = (W, F_a \cup F_b)$ (where each edge has weight γ). We can observe its behavior easily by clustering node pair w^a_i and w^b_i for each $i \in [0, N-1]$. Figure 2 gives an alternative drawing of $\Gamma_{H,\gamma}(F_a, F_b)$ for the instance shown in Figure 1. Each light-gray band corresponds to an edge e in H, which contains at least one actual path of length γ if and only if $e \in F_a$ or $e \in F_b$ holds. For this construction, we can show the following lemma:

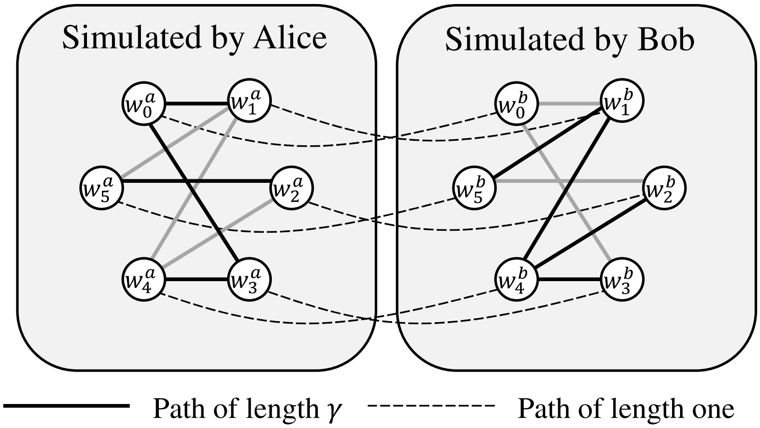

Fig. 2. Construction of $\Gamma_{H,\gamma}(E_a, E_b)$ for the disjoint instance in Figure 1

Lemma 1. *Let (F_a, F_b) be an instance of the graphic set-disjointness over $H = (W, F)$, $H' = (W, F_a \cup F_b)$, and $\Gamma = \Gamma_{H,\gamma}(F_a, F_b)$ for short. Then, for any integer $k > 0$, the following two properties hold:*

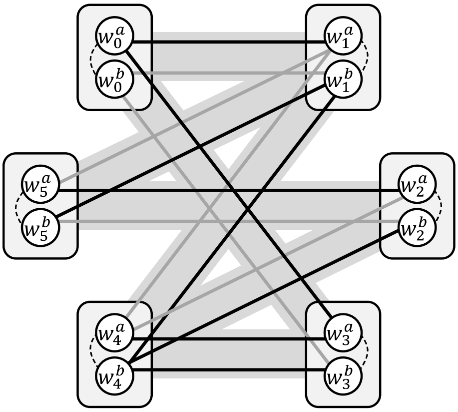

Fig. 3. An alternative view of $\Gamma_{H,\gamma}(E_a, E_b)$ shown in Figure 2

– *If* $d_{H'}(w_i, w_j) = 1$ $(i \neq j)$, $d_\Gamma(w_i^a, w_j^a) \leq \gamma + 2$.
– *If* $d_{H'}(w_i, w_j) = k$ $(k > 1, i \neq j)$, $d_\Gamma(w_i^a, w_j^a) \geq k\gamma$.

Proof. If $d_{H'}(w_i, w_j) = 1$, $(w_i, w_j) \in F_a$ or $(w_i, w_j) \in F_b$ holds. It implies that Γ contains an inter-cluster path between w_i^a and w_j^a or w_i^b and w_j^b. Since Γ always contains the edges (w_i^a, w_i^b) and (w_j^b, w_j^b), the first property obviously holds.

We look at the second property. Suppose for contradiction that there exists a simple path P between w_i^a and w_j^a whose length is less than $k\gamma$. Note that P is a concatenation of several inter-cluster paths and intra-cluster paths. It implies that P contains at most $k-1$ inter-cluster paths. Now let P' be the path obtained from P by contracting all the intra-cluster paths. Since P' is the concatenation of several inter-cluster paths, it can be represented by some sequence of the nodes where two inter-cluster paths are concatenated. Let $w_{\beta_0}^{\alpha_0}, w_{\beta_1}^{\alpha_1}, \cdots, w_{\beta_l}^{\alpha_l}$ be that sequence $(0 < l \leq k-1)$. Then, for any $x \in [0, l-1]$, $w_{\beta_x}^{\alpha_x}$ and $w_{\beta_{x+1}}^{\alpha_{x+1}}$ must be connected by an inter-cluster path. That is, either $(w_{\beta_x}, w_{\beta_{x+1}}) \in F_a$ or $(w_{\beta_x}, w_{\beta_{x+1}}) \in F_b$ must hold. However, it implies that $H' = (W, F_a \cup F_b)$ contains a path from w_i to w_j with length $l (< k)$. It is a contradiction. □

The main theorem utilizes the conjecture below:

Conjecture 1 (Girth conjecture). For any integers N and t, there exists a graph $H_{t,N}$ of N nodes and $\Theta(N^{1+1/t})$ edges whose girth is at least $2t + 2$.

Theorem 2. *Assume that the girth conjecture is true for some constant* $t > 0$. *Let ALG be a distributed algorithm constructing distance oracles with stretch* $2t$. *Then, its worst-case running time* $\tau(n)$ *must satisfy* $\tau(n) \geq \Omega\left(n^{\frac{1}{t+1}} / \log n\right)$.

Proof. The theorem is proved by the reduction from the graphic set-disjointness over $H_{t,N}$ claimed in Conjecture 1 (the value of N is determined later). That is, we construct from ALG a two-party protocol solving the graphic set-disjointness problem over $H_{t,N}$ for any instance (F_a, F_b). The core of the construction is to simulate the run of ALG in $\Gamma_{H_{t,N},8t}(F_a, F_b)$. Let $\Gamma = \Gamma_{H_{t,N},8t}(F_a, F_b)$ for short. Alice simulates all the processes in W^a and Bob those in W^b. To make the simulation proceed, both Alice and Bob need to obtain the messages exchanged on intra-cluster paths in the run of ALG. Since there are N intra-cluster paths, the amount of the information transmitted through the paths is at most $O(N \log n)$ bits per one round. Thus to complete the simulation, it suffices that Alice and Bob totally exchange $O(\tau(n)N \log n)$-bit messages. After the simulation, Alice checks the distance of each pair $(w^a_i, w^a_j) \in F$ by querying it to w^a_i's local oracle. Note that this query is locally processed at Alice. From Lemma 1, if $(w^a_i, w^a_j) \in F_a \cup F_b$ holds, the distance between w^a_i and w^a_j in Γ is at most $8t+2$ (remind $\gamma = 8t$). Hence the distance estimated by the oracle is at most $2t(8t+2)$. On the other hand, if $(w_i, w_j) \notin F_a \cup F_b$, the distance between w_i and w_j in the graph $H_{t,N} \setminus (w_i, w_j)$ is at least $2t + 1$ because the girth of $H_{t,N}$ is at least $2t+2$. Thus, by Lemma 1, the distance between w^a_i and w^b_j is at least $8t(2t+1)$. These two facts imply that Alice can determine the disjointness of (F_a, F_b) from the query results: If all the queries return values at most $2t(8t + 2)$, (F_a, F_b) is disjoint. Otherwise, it is intersecting. Finally Alice sends one-bit information of the decision.

The two-party protocol explained above totally consumes $O(\tau(n)N \log n)$ bits in the worst case, which must be lower bounded by the communication complexity of the graphic set-disjointness over $H_{t,N}$, that is, $\Omega(N^{1+1/t})$ bits. Now we rewrite variable N by using only n and t. Since the number n of nodes in $\Gamma_{H_{t,N},8t}(F_a, F_b)$ is $2N + (8t - 1) \cdot \Theta(N^{1+1/t}) = \Theta(tN^{1+1/t})$, $N = \Theta((n/t)^{t/(t+1)})$ holds. Since t is a constant, we have $N = \Theta(n^{t/(t+1)})$. Thus the total amount of messages exchanged by the proposed two-party protocol is $\Theta((n^{t/(t+1)} \cdot (\tau(n) \log n))$. Since this is bounded by $\Omega(N^{1+1/t}) = \Omega(n)$. It follows that $\tau(n) = \Omega(n^{1/(t+1)}/ \log n)$. The theorem is proved. □

4 Lower Bound for Weighted Graphs

The lower bound in the previous section is extended to a stronger lower bound for weighted graphs. The fundamental idea of the extension is to utilize the framework by Das Sarma et al. [2]. Given values N and t, let $N^- = N^{\frac{1}{2}-\frac{1}{5t}}$ and $N^+ = N^{\frac{1}{2}+\frac{1}{5t}}$ for short. For simplicity, we assume that N^+ is a power of two. Note that this assumption is not essential and easily removed without affecting the asymptotic complexity we prove in this section. The gadget graph $\Gamma'_H(F_a, F_b)$ (say Γ' for short) is built by the following steps:

1. We first prepare N^- paths of length N^+, each of which is denoted by P_i $(0 \leq i \leq N^- - 1)$. The nodes constituting P_i are identified by $v_{(i,0)}, v_{(i,1)}, \cdots, v_{(i,N^+-1)}$ from left to right. The weight of each edge constituting these paths is one. Furthermore, we give an alias to each endpoint node. We refer to nodes $v_{(i,0)}$ and $v_{(i,N^+-1)}$ as w^a_i and w^b_i respectively $(0 \leq i \leq N^- - 1)$. We also define $W^a = \{w^a_0, w^a_1, \cdots w^a_{N^- -1}\}$ and $W^b = \{w^b_0, w^b_1, \cdots w^b_{N^- -1}\}$.

2. Construct a complete binary tree with N^+ leaves. The leaf nodes in the tree are labeled by $u_0, u_1, \cdots u_{N^+-1}$ from left to right. The weight of edges in the tree is $100N^+N^-t^2$.
3. Add edges $(u_i, v_{(j,i)})$ for any $i \in [0, N^+ - 1]$ and $j \in [0, N^- - 1]$. These edges also has weight $100N^+N^-t^2$.
4. Encode the instance (F_a, F_b) to the graph induced by W^a and W^b. That is, an edge (w_i^a, w_j^a) (resp. (w_i^b, w_j^b)) is connected by an edge of weight $8tN^+$ if and only if $(w_i, w_j) \in F_a$ (resp. $(w_i, w_j) \in F_b$).

The whole construction is illustrated in Figure 4. Note that the number n of nodes in $\Gamma'_H(F_a, F_b)$ is $\Theta(N)$, and its diameter is $D = O(\log n)$. This gadget has a structure similar to the unweighted case. We have the following lemma:

Lemma 2. *Let (F_a, F_b) be an instance of the graphic set-disjointness problem over $H = (W, F)$, and $H' = (W, F_a \cup F_b)$ for short. Then, for any integer $k > 0$, the following two properties hold:*

- *If $d_{H'}(w_i, w_j) = 1$ $(i \neq j)$, $d_{\Gamma'}(w_i^a, w_j^a) \leq (8t + 2)N^+$.*
- *If $d_{H'}(w_i, w_j) = k$ $(k > 1,\ i \neq j)$, $d_{\Gamma'}(w_i^a, w_j^a) \geq 8tkN^+$.*

Proof. Since all the edges augmented in Step 2 and 3 of the construction are too heavy, they are not contained in the shortest path between w_i^a and w_j^b for any i and j. Thus we can omit those edges in the proof (in Figure 4, they are grayed out). Then, the graph $\Gamma' = \Gamma'_H(F_a, F_b)$ can be seen as a weighted version of $\Gamma = \Gamma_{H,8t}(F_a, F_b)$: The length of the path between w_i^a and w_i^b $(0 \leq i \leq N^- - 1)$ is N^+ (which corresponds to intra-cluster paths in Γ) and each edge between $(w_i^a, w_j^a) \in F_a$ (resp. $(w_i^b, w_j^b) \in F_b$) has weight $8tN^+$. That is, we have $d_{\Gamma'}(w_i^a, w_j^a) = N^+ \cdot d_{\Gamma}(w_i^a, w_j^a)$. Consequently, the lemma is deduced from Lemma 1. □

The following theorem is the core of the reduction.

Theorem 3 (Das Sarma et al. [2]). *Let ALG be any algorithm running on the graph Γ', where H is an arbitrary graph of N^- nodes. Then there exists a two-party protocol satisfying the following three properties:*

- *At the beginning of the protocol, Alice (resp. Bob) knows the whole topological information of Γ' except for the subgraph induced by W^b (resp. W^a),*
- *after the run of the protocol, Alice and Bob output the internal states of the processes in W^a and W^b at round $N^+/2$ in the execution of ALG, respectively, and*
- *the protocol consumes at most $O(N^+(\log n)^2)$-bit communication.*

While the graph used in this paper is a slightly modified version of the original construction in [2], the theorem above is proved in the almost same way. So we just quote it without the proof.

The theorem above induces our lower bound via a reduction from two-party graphic set-disjointness:

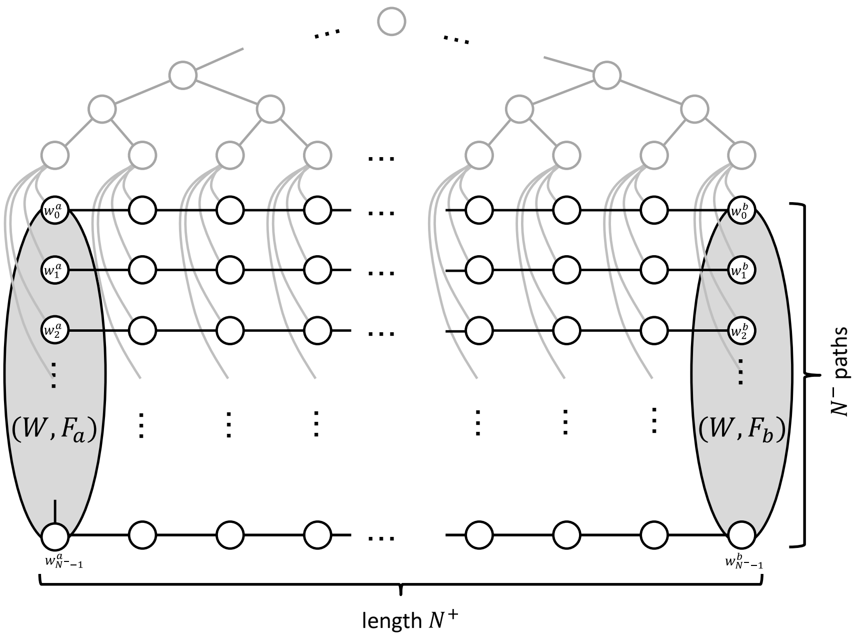

Fig. 4. Construction of $\Gamma'_H(F_a, F_b)$

Theorem 4. *Assume that the girth conjecture is true for some constant $t > 0$. Let ALG be an algorithm constructing distributed (weighted) distributed distance oracles with stretch $2t$. Then, its worst-case running time $\tau(n)$ must satisfy $\tau(n) = \Omega(N^+) = \Omega\left(n^{\frac{1}{2}+\frac{1}{5t}}\right)$.*

Proof. The proof is almost the same as Theorem 2. Letting H_{t,N^-} be the graph claimed in Conjecture 1 and (F_a, F_b) be any instance of the graphic set-disjointness over H_{t,N^-}, we consider the run of ALG in the graph $\Gamma'_{H_{t,N^-}}(F_a, F_b)$. Suppose for contradiction that $\tau(n) < N^+/2$ holds. Then, by Theorem 3, we can have a two-party protocol where Alice and Bob simulate the run of ALG at the processes in W^a and W^b respectively. After the simulation, Alice queries the distance between w_i^a and w_j^b for each $(w_i, w_j) \in F$. Then, by Lemma 2 and the same argument as the proof of Theorem 2, Alice can determine the disjointness of (F_a, F_b). That is, if all the queries return values at most $2t(8t+2)N^+$, (F_a, F_b) is disjoint. Otherwise, it is intersecting. Finally Alice sends the one-bit information of the decision. By Theorem 3, this protocol consumes only $O(N^+(\log n)^2)$-bit communication for deciding the disjointness of (F_a, F_b). However, the communication complexity of the graphic set-disjointness over H_{t,N^-} is bounded by the number of edges in the base graph H_{t,N^-}. That is, from Conjecture 1, it is lower bounded by $\Omega((N^-)^{(1+1/t)}) = \Omega((N^{\frac{1}{2}-\frac{1}{5t}})^{(1+1/t)}) = \Omega(N^{\frac{1}{2}+\frac{1}{5t}+\epsilon}) = \omega(N^+(\log n)^2)$, where ϵ is a small constant (depending on t). It is a contradiction. $\square$

5 Lower Bound for Bounded Label Size Oracles

In this section, we present an unconditional lower bound for the case of bounded label size oracles. For lack of space, we only focus on the bound for unweighted graphs, but its result is easily extended to the weighted case by combining the argument in Section 4.

The fundamental idea follows the proof in Section 3. We construct a reduction from the two-party graphic set-disjointness. The main difference is that we use a graph of $\Theta(N^{1+1/t-\epsilon})$ nodes with girth $(2t+2)$ as the base graph (where ϵ is a small constant depending on t), but augment only N intra-cluster paths crossing Alice and Bob sides. The following lemma is an alternative to the girth conjecture.

Lemma 3. *Let $\epsilon \leq 1/2t^2$. For any sufficiently large integer N, there exists a bipartite graph $H = (U \cup W, F)$ such that $|U| = N^{1+1/t-\epsilon}$, $|W| = N$, and $|F| = N^{1+1/t}$ hold and the girth is at least $2t+2$.*

Proof. The proof idea is based on the seminal one by Erdos's probabilistic method, which shows an existence of the graph with high chromatic number and girth [3,4]. We consider the random construction of a bipartite graph H^* whose node set is $U \cup W$. That is, fixing the vertex set U and W, for each pair $(u, w) \in U \times W$, we add an edge with probability $1/N^{1-\epsilon}$. Then the graph H^* satisfies the following two properties with a non-zero probability: (1) The number of edges is $\Omega(N^{1+1/t})$, and (2) there are only $o(N^{1+1/t})$ cycles with a length less than or equal to $2t$. Once we find a graph H^* with both properties, the desired graph H is obtained from H^* by removing $o(N^{1+1/t})$ edges from each short cycle, which still have $\Omega(N^{1+1/t})$ edges but there is no cycle with length less than $2t+2$ (remind that the graph is bipartite and thus there is no cycle of length $2t+1$). Thus the remaining part of the proof is to show that the properties (1) and (2) are simultaneously satisfied with a non-zero probability. More precisely, it suffices to show that each property is satisfied with a probability more than 1/2. Then using the union bound, the probability that either property (1) or (2) fails becomes strictly smaller than one.

The first property is almost trivial. Let X be the number of edges in H'. Since the variable X is the sum of independent Poisson trials, we can apply Chernoff bounds [10]. Then it is not difficult to obtain $\Pr[X < E[X]/2] = o(1)$. That is, $X \geq E[X]/2$ holds with probability more than 1/2. The expected number $E[X]$ of edges in H' is $|U||W| \cdot (1/N^{1-\epsilon}) = N^{1+1/t}$, and thus the first property holds.

We look at the second property. Let Y be the number of cycles with length less than $2t+2$ in H'. The probability that a given sequence of $2k$ nodes ($k \leq t$) form a cycle is obviously bounded by $(1/N^{1-\epsilon})^{2k}$. Since we assume $\epsilon \leq 1/2t^2$, the expected number $E[Y]$ is bounded as follows:

$$E[Y] = \sum_{k=1}^{t} \binom{N^{1+\frac{1}{t}-\epsilon}}{k} \binom{N}{k} \cdot \left(\frac{1}{N^{1-\epsilon}}\right)^{2k}$$
$$\leq tN^{t(1+\frac{1}{t}-\epsilon)} N^t \cdot N^{-2t(1-\epsilon)}$$
$$\leq tN^{1+\epsilon t}$$
$$\leq O(N^{1+1/2t}),$$

Using Markov's inequality [10], we can have

$$\Pr[Y \geq 3E[Y]] \leq E[Y]/(3E[Y]) = 1/3.$$

Thus the property (2) is also satisfied with probability more than 1/2. The lemma is proved. □

Let $H = (U \cup W, F)$ be the graph proposed in Lemma 3 for $\epsilon = 1/2t^2$. The gadget graph $\hat{\Gamma}_{H,8t}(F_a, F_b)$ to encode the graphic set-disjointness (F_a, F_b) over H is constructed similarly to $\Gamma_{H,8t}(F_a, F_b)$ in Section 3. Only the difference is that we connect Alice and Bob sides only by edges (w_i^a, w_i^b) for any i $(0 \leq i \leq N-1)$, but not connect u_i^a and u_i^b. The constructed gadget is presented in Figure 5. We define $U^a = \{u_0^a, u_1^a, \cdots, u_{N-1}^a\}$ and $U^b = \{u_0^b, u_1^b, \cdots, u_{N-1}^b\}$.

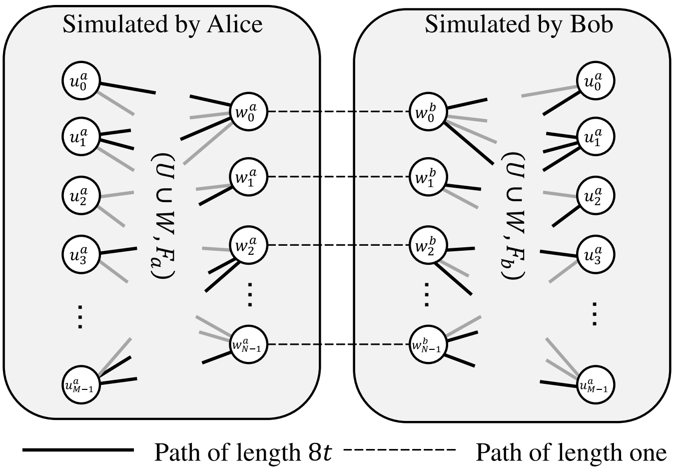

Fig. 5. Construction of $\hat{\Gamma}_{H,8t}(F_a, F_b)$

For this construction, we can have a lemma analogous to Lemma 1.

Lemma 4. *Let (F_a, F_b) be an instance of the graphic set-disjointness over H, $H' = (U, W, F_a \cup F_b)$, and $\hat{\Gamma} = \hat{\Gamma}_{H,8t}(F_a, F_b)$ for short. Then, for any integer $k > 0$, the following two properties hold:*

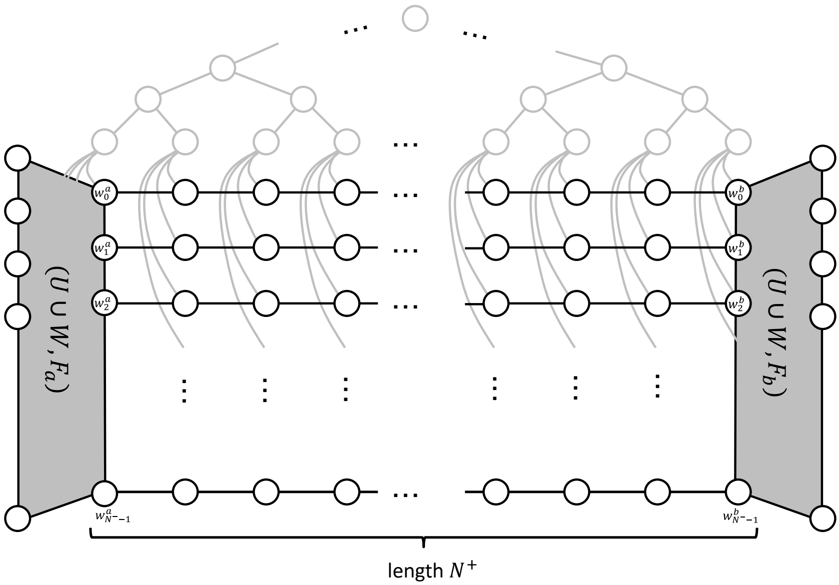

Fig. 6. Construction of the gadget for weighted and bounded label-size oracles

– *If* $(w_i, u_j) \in F_b$*,* $d_{\hat{\Gamma}}(w_i^a, u_j^b) \leq (8t + 1)$*.*
– *If* $(w_i, u_j) \notin F_a \cup F_b$*,* $d_{\hat{\Gamma}}(w_i^a, u_j^b) \geq 8t(2t + 1)$*.*

While the proof is omitted for lack of space, it is almost the same as that for Lemma 1. We show the main theorem:

Theorem 5. *Let* ALG *be an algorithm constructing distributed distance oracles with stretch* $2t$ *and* $o(n^{1/2t(t+1)})$*-bit label size. Then, its worst-case running time* $\tau(n)$ *must satisfy* $\tau(n) \geq \Omega\left(n^{\frac{1}{t+1}} / \log n\right)$*.*

Proof. The proof basically follows that for Theorem 2. To construct a two-party graphic set-disjointness protocol, Alice and Bob simulate the internal states of the processes W^a and U^a, and W^b and U^b in the run of ALG, respectively. After the simulation, since Bob knows all the labels assigned to the nodes in U^b, it sends them to Alice. This information allows Alice to estimate the distance between w_i^a and u_j^b for any i and j *locally*. Then, by Lemma 4, Alice can determine the existence of the edge (w_i^b, u_j^b) for i, j such that $(w_i^a, u_j^a) \notin F_a$ holds. That is, Alice first queries the distance between w_i^a and u_j^b, and then if the estimated distance is less than or equal to $2t(8t + 1)$, it decides $(w_i^a, u_j^b) \in F_b$. Repeating this kind of queries, Alice can determines the disjointness of (F_a, F_b).

Compared to the protocol proposed in the proof of Theorem 2, the extra communication incurred by this protocol is to send the labels of the nodes in U^b from Bob to Alice. Since the label size for one node is $o(n^{1/2t(t+1)}) = o(N^{1/2t^2})$

bits, the amount of the extra communication is $o(N^{1/2t^2}) \cdot |U^b| = o(N^{1+1/t})$ bits, which is not a dominant part of the protocol communication because solving the graphic set-disjointness requires $\Omega(N^{1+1/t})$-bit communication. Consequently, the amount of the communication spent for the simulation must be $\Omega(N^{1+1/t})$ bits, and thus we have the same bound for $\tau(n)$ as Theorem 2. □

By applying the same approach, we can also obtain the lower bound for weighted graphs. The gadget construction is illustrated in Figure 6. The encoding of H is similar with the construction of $\hat{\Gamma}$. For Alice (resp. Bob) side, only the nodes in W^a (resp. W^b) overlap the endpoints of the paths. Following the arguments in Theorem 4 and 5, we can show the theorem below:

Theorem 6. *Let ALG be a distributed algorithm constructing weighted distance oracles with stretch $2t$ and and $o(n^{1/5t^2})$-bit label size. Then, its worst-case running time $\tau(n)$ must satisfy $\tau(n) = \Omega\left(n^{\frac{1}{2}+\frac{1}{5t}}\right)$.*

6 Conclusion

We presented time lower bounds for the distributed distance oracle construction. Our primary result is to exhibit a trade-off between construction time and stretch. More precisely, given stretch factor $2t$, our lower bounds have the form of $\tilde{\Omega}(n^{1/O(t)})$ rounds for unweighted graphs, and the form of $\tilde{\Omega}(n^{1/2+1/O(t)})$ rounds for weighted graphs. While we assume that the girth conjecture is true for proving the bounds, we can bypass it when we consider bounded label-size ofracles. Restricting the label size to n^{ϵ} for a small constant ϵ depending on t, the same lower bounds are unconditionally obtained. An open problem related to our results is to find algorithms whose running time gets close to our lower bounds. The currently best algorithm in weighted graphs takes $O(n^{\frac{1}{2}+\frac{1}{2t}})$ rounds for the construction and achieves $O(t^2)$ stretch. The algorithm whose stretch linearly depends on t but achieving $O(n^{\frac{1}{2}+\frac{1}{O(t)}})$-round construction time is still open. Faster solutions for unweighted graphs are also not known.

References

1. Das Sarma, A., Dinitz, M., Pandurangan, G.: Efficient computation of distance sketches in distributed networks. In: Proc. of the 24th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA), pp. 318–326 (2012)
2. Das Sarma, A., Holzer, S., Kor, L., Korman, A., Nanongkai, D., Pandurangan, G., Peleg, D., Wattenhofer, R.: Distributed verification and hardness of distributed approximation. In: Proc. of the 43rd Annual ACM Symposium on Theory of Computing, pp. 363–372 (2011)
3. Diestel, R.: Graph Theory, 4th edn., vol. 173. Springer (2012)
4. Erdös, P.: Graph theory and probability. In: Classic Papers in Combinatorics, pp. 276–280 (1987)
5. Frischknecht, S., Holzer, S., Wattenhofer, R.: Networks cannot compute their diameter in sublinear time. In: Proc. of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 1150–1162 (2012)

6. Holzer, S., Wattenhofer, R.: Optimal distributed all pairs shortest paths and applications. In: Proc. of the 2012 ACM Symposium on Principles of Distributed Computing (PODC), pp. 355–364 (2012)
7. Kalyanasundaram, B., Schintger, G.: The probabilistic communication complexity of set intersection. SIAM Journal on Discrete Mathematics 5(4), 545–557 (1992)
8. Lenzen, C., Patt-Shamir, B.: Fast routing table construction using small messages: Extended abstract. In: Proc. of the 45th Annual ACM Symposium on Symposium on Theory of Computing (STOC), pp. 381–390 (2013)
9. Lenzen, C., Peleg, D.: Efficient distributed source detection with limited bandwidth. In: Proc. of the 2013 ACM Symposium on Principles of Distributed Computing (PODC), pp. 375–382 (2013)
10. Mitzenmacher, M., Upfal, E.: Probability and Computing: Randomized Algorithms and Probabilistic Analysis. Cambridge University Press (2005)
11. Nanongkai, D.: Distributed approximation algorithms for weighted shortest paths. In: Proc. of the 46th Annual ACM Symposium on Theory of Computing (STOC), pp. 565–573 (2014)
12. Peleg, D., Roditty, L., Tal, E.: Distributed algorithms for network diameter and girth. In: Czumaj, A., Mehlhorn, K., Pitts, A., Wattenhofer, R. (eds.) ICALP 2012, Part II. LNCS, vol. 7392, pp. 660–672. Springer, Heidelberg (2012)
13. Razborov, A.A.: On the distributional complexity of disjointness. Theoretical Computer Science 106(2), 385–390 (1992)
14. Thorup, M., Zwick, U.: Approximate distance oracles. Journal of the ACM 52(1), 1–24 (2005)
15. Yao, A.C.-C.: Some complexity questions related to distributive computing (preliminary report). In: Proc. of the 11th Annual ACM Symposium on Theory of Computing (STOC), pp. 209–213 (1979)