Optimal Lower Bounds for Universal and Differentially Private Steiner Trees and TSPs

Anand Bhalgat[*], Deeparnab Chakrabarty, and Sanjeev Khanna[**]

Department of Computer and Information Science, University of Pennsylvania
{bhalgat,deepc}@seas.upenn.edu, sanjeev@cis.upenn.edu

Abstract. Given a metric space on n points, an α-approximate *universal* algorithm for the Steiner tree problem outputs a distribution over rooted spanning trees such that for any subset X of vertices containing the root, the expected cost of the induced subtree is within an α factor of the optimal Steiner tree cost for X. An α-approximate *differentially private* algorithm for the Steiner tree problem takes as input a subset X of vertices, and outputs a tree distribution that induces a solution within an α factor of the optimal as before, and satisfies the additional property that for any set X' that differs in a single vertex from X, the tree distributions for X and X' are "close" to each other. Universal and differentially private algorithms for TSP are defined similarly. An α-approximate universal algorithm for the Steiner tree problem or TSP is also an α-approximate differentially private algorithm. It is known that both problems admit $O(\log n)$-approximate universal algorithms, and hence $O(\log n)$ approximate differentially private algorithms as well.

We prove an $\Omega(\log n)$ lower bound on the approximation ratio achievable for the universal Steiner tree problem and the universal TSP, matching the known upper bounds. Our lower bound for the Steiner tree problem holds even when the algorithm is allowed to output a more general solution of a distribution on paths to the root. We then show that whenever the universal problem has a lower bound that satisfies an additional property, it implies a similar lower bound for the differentially private version. Using this converse relation between universal and private algorithms, we establish an $\Omega(\log n)$ lower bound for the differentially private Steiner tree and the differentially private TSP. This answers a question of Talwar [19]. Our results highlight a natural connection between universal and private approximation algorithms that is likely to have other applications.

1 Introduction

Traditionally, in algorithm design one assumes that the algorithm has complete access to the input data which it can use unrestrictedly to output the optimal, or near optimal, solution. In many applications, however, this assumption does not hold and the traditional approach towards algorithms needs to be revised. For

[*] Supported in part by NSF Awards CCF-0635084 and IIS-0904314.

L.A. Goldberg et al. (Eds.): APPROX/RANDOM 2011, LNCS 6845, pp. 75–86, 2011.

instance, let us take the problem of designing the cheapest multicast network connecting a hub node to a set of client nodes; this is a standard network design problem which has been studied extensively. Consider the following two situations. In the first setting, the actual set of clients is unknown to the algorithm, and yet the output multicast network must be "good for all" possible client sets. In the second setting, the algorithm knows the client set, however, the algorithm needs to ensure that the output preserves the privacy of the clients. Clearly, in both these settings, the traditional algorithms for network design don't suffice.

The situations described above are instances of two general classes of problems recently studied in the literature. The first situation needs the design of *universal* algorithms; algorithms which output solutions when parts of the input are uncertain or unknown. The second situation needs the design of *differentially private* algorithms; algorithms where parts of the input are controlled by clients whose privacy concerns constrain the behaviour of the algorithm. A natural question arises: how do the constraints imposed by these classes of algorithms affect their performance?

In this paper, we study universal and differentially private algorithms for two fundamental combinatorial optimization problems: the Steiner tree problem and the travelling salesman problem (TSP). The network design problem mentioned above corresponds to the Steiner tree problem. We resolve the performance question of universal and private algorithms for these two problems completely by giving lower bounds which match the known upper bounds. Our techniques and constructions are quite basic, and we hope these could be applicable to other universal and private algorithms for sequencing and network design problems.

Problem formulations. In both the Steiner tree problem and the TSP, we are given a metric space (V, c) on n vertices with a specified root vertex $r \in V$. Given a subset of terminals, $X \subseteq V$, we denote the cost of the optimal Steiner tree connecting $X \cup r$ by $\mathtt{opt}_{ST}(X)$. Similarly, we denote the cost of the optimal tour connecting $X \cup r$ by $\mathtt{opt}_{TSP}(X)$. If X is known, then both $\mathtt{opt}_{ST}(X)$ and $\mathtt{opt}_{TSP}(X)$ can be approximated up to constant factors.

A *universal* algorithm for the Steiner tree problem, respectively the TSP, does not know the set of terminals X, but must output a distribution $\mathcal{D}$ on rooted trees T, respectively tours σ, spanning all vertices of V. Given a terminal set X, let $T[X]$ be the minimum-cost rooted subtree of T which contains X. Then the cost of the universal Steiner tree algorithm on terminal set X is $\mathbf{E}_{T \leftarrow \mathcal{D}}[c(T[X])]$. We say the universal Steiner tree algorithm is *α-approximate*, if for all metric spaces and all terminal sets X, this cost is at most $\alpha \cdot \mathtt{opt}_{ST}(X)$. Similarly, given a terminal set X, let σ_X denote the order in which vertices of X are visited in σ, and let $c(\sigma_X)$ denote the cost of this tour. That is, $c(\sigma_X) := c(r, \sigma_X(1)) + \sum_{i=1}^{|X|-1} c(\sigma_X(i), \sigma_X(i+1)) + c(\sigma_X(|X|), r)$. The cost of the universal TSP algorithm on set X is $\mathbf{E}_{T \leftarrow \mathcal{D}}[c(\sigma_X)]$, and the approximation factor is defined as it is for the universal Steiner tree algorithm.

A *differentially private* algorithm for Steiner trees and TSPs, on the other hand, knows the set of terminals X; however, there is a restriction on the solution that it can output. Specifically, a differentially private algorithm for the Steiner tree problem with privacy parameter ε, returns on any input terminal set X a distribution $\mathcal{D}_X$ on trees spanning V, with the following property. Let X' be any terminal set such that the symmetric difference of X' and X is exactly one vertex. Then,

$$\Pr_{\mathcal{D}_{X'}}[T] \cdot \exp(-\varepsilon) \;\leq\; \Pr_{\mathcal{D}_X}[T] \;\leq\; \Pr_{\mathcal{D}_{X'}}[T] \cdot \exp(\varepsilon),$$

where $\Pr_{\mathcal{D}}[T]$ is the probability of getting tree T when drawn from distribution $\mathcal{D}$. The cost of the algorithm on set X is $\mathbf{E}_{T \leftarrow \mathcal{D}_X}[c(T[X])]$ as before, and the approximation factor is defined as that for universal trees. Differentially private algorithms for the TSP are defined likewise. To gain some intuition as to why this definition preserves privacy, suppose each vertex is a user and controls a bit which reveals its identity as a terminal or not. The above definition ensures that even if a user changes its identity, the algorithm's behaviour does not change by much, and hence the algorithm does not leak any information about the user's identity. This notion of privacy is arguably the standard and strongest notion of privacy in the literature today; we point the reader to [4] for an excellent survey on the same. We make two simple observations; (a) any universal algorithm is a differentially private algorithm with $\varepsilon = 0$, (b) if the size of the symmetric difference in the above definition is k instead of 1, then one can apply the definition iteratively to get $k\varepsilon$ in the exponent.

For the Steiner tree problem, one can consider another natural and more general solution space for universal and private algorithms, where instead of returning a distribution on trees spanning V, the algorithm returns a distribution $\mathcal{D}$ on collections of paths $P := \{p_v : v \in V\}$, where each p_v is a path from v to the root r. Given a single collection P, and a terminal set X, the cost of the solution is $c(P[X]) := c(\big(\bigcup_{v \in X} E(p_v)\big)$, where $E(p_v)$ is the set of edges in the path p_v. The cost of the algorithm on set X is $\mathbf{E}_{P \leftarrow \mathcal{D}}[c(P[X])]$. Since any spanning tree induces an equivalent collection of paths, this solution space is more expressive, and as such, algorithms in this class may achieve stronger performance guarantees. We show that this more general class of algorithms has the same asymptotic lower bound as the class of algorithms that are restricted to output a spanning tree.

1.1 Previous Work and Our Results

A systematic study of universal algorithms was initiated by Jia et al. [12], who gave $O(\log^4 n / \log\log n)$-approximate universal algorithms for both the Steiner tree problem and the TSP. Their algorithms were deterministic and returned a single tree and tour respectively. The authors also noted that results of [2,5] on probabilistically embedding general metrics into tree metrics imply randomized $O(\log n)$-approximate universal algorithms for these problems. Using properties of the embeddings of [5], Gupta et al.[7] gave deterministic $O(\log^2 n)$-approximate universal algorithms for both problems.

Jia et al. [12] observe that a lower bound for online Steiner tree algorithms implies a lower bound for universal Steiner tree algorithms; thus, following the result of Imase and Waxman [11], one obtains a lower bound of $\Omega(\log n)$ for any universal Steiner tree algorithm. It is not hard to see that the [11] lower bound also holds for algorithms returning a collection of vertex-to-root paths. Jia et al. [12] explicitly leave lower bounds for the universal TSP as an open problem. Hajiaghayi et al. [9] make progress on this by showing an $\Omega\left(\sqrt[6]{\log n/\log\log n}\right)$ lower bound for universal TSP; this holds even in the two dimensional Euclidean metric space. [9] conjecture that for general metrics the lower bound should be $\Omega(\log n)$; in fact, they conjecture this for the shortest path metric of a constant degree expander. Very recently, this conjecture was proven by Gorodezky et al. [6]; we discuss and compare this particular result and ours at the end of this subsection.

When the metric space has certain special properties (for instance if it is the Euclidean metric in constant dimensional space), Jia et al. [12] give an improved universal algorithms for both Steiner tree and TSP, which achieves an approximation factor of $O(\log n)$ for both problems. Furthermore, if the size of the terminal set X is k, their approximation factor improves to $O(\log k)$ – a significant improvement when $k \ll n$. This leads to the question whether universal algorithms exist for these problems whose approximation factors are a non-trivial function of k alone. A k-approximate universal Steiner tree algorithm is trivial; the shortest path tree achieves this factor. This in turn implies a $2k$-approximate universal TSP algorithm. Do either of these problems admit an $o(k)$-approximate algorithm? The constructions of [11] achieving a lower bound of $\Omega(\log n)$ for universal Steiner tree require terminal sets that are of size $n^{\Omega(1)}$, and do not rule out the possibility of an $O(\log k)$-approximation in general. In fact, for many network optimization problems, an initial polylog(n) approximation bound was subsequently improved to a polylog(k) approximation (e.g., sparsest cut [13,14], asymmetric k-center [18,1], and more recently, the works of Moitra et al. [16,17] on vertex sparsifiers imply such a result for other many cut and flow problems). It is thus conceivable that a polylog(k)-approximation could be possible for the universal algorithms as well.

We prove $\Omega(\log n)$ lower bounds for the universal TSP and the Steiner tree problem, even when the algorithm returns vertex-to-root paths for the latter (Theorems 2 and 1). Furthermore, the size of the terminal sets in our lower bounds is $\Theta(\log n)$, ruling out any $o(k)$-universal algorithm for either of these problems.

Private vs universal algorithms. The study of differentially private algorithms for combinatorial optimization problems is much newer, and the paper by Gupta et al. [8] gives a host of private algorithms for many optimization problems. Since any universal algorithm is a differentially private algorithm with $\varepsilon = 0$, the above stated upper bounds for universal algorithms hold for differentially private algorithms as well. For the Steiner tree problem and TSP, though, no better differentially private algorithms are known. Talwar, one of the authors of [8], recently posed an open question whether a private $O(1)$-approximation exists for the Steiner tree problem, even if the algorithm is allowed to use a more general solution space, namely, return a collection of vertex-to-root paths, rather than Steiner trees [19].

We observe that a simple but useful converse relation holds between universal and private algorithms: "strong" lower bounds for universal algorithms implies lower bounds for differentially private algorithms. More precisely, suppose we can show that for any universal algorithm for the Steiner tree problem/TSP, there exists a terminal set X, such that the probability that a tree/tour drawn from the distribution has cost less than α times the optimal cost is $\exp(-\varepsilon|X|)$ for a certain constant ε. *Then we get an $\Omega(\alpha)$ lower bound on the performance of any ε-differentially private algorithm for these problems.* (Corollary 1). Note that this is a much stronger statement than merely proving a lower bound on the expected cost of a universal algorithm. The expected cost of a universal algorithm may be $\Omega(\alpha)$, for instance, even if it achieves optimal cost with probability 1/2, and α times the optimal cost with probability 1/2. In fact, none of the earlier works mentioned above [11,12,9,6] imply strong lower bounds. The connection between strong lower bound on universal algorithms and lower bounds for differentially private algorithms holds for a general class of problems, and may serve as a useful tool for establishing lower bounds for differentially private algorithms (Section 3).

All the lower bounds we prove for universal Steiner trees and TSP are strong in the sense defined above. As corollaries, we get lower bounds of $\Omega(\log n)$ on the performance of differentially private algorithms TSP and the Steiner tree problem, even when the algorithm returns a collection of paths. This answers the question of Talwar [19] negatively. (Corollaries 1 and 2).

The metric spaces for our lower bounds on universal Steiner tree and TSP are shortest path metrics on constant degree expanders. To prove the strong lower bounds on distributions of trees/tours, it suffices, by Yao's lemma, to construct a distributions on terminal sets such that any fixed tree/tour pays, with high probability, an $\Omega(\log n)$ times the optimum tree/tour's cost on a terminal set picked from the distribution. We show that vertices on a sufficiently long random walk suffices in the Steiner tree case, while for TSP, we choose the client set from two independent random walks.

Comparison of our results with [6]: As mentioned above, Gorodezky et al. [6] obtain an $\Omega(\log n)$ lower bound for universal TSP. Their result also gives an $\Omega(k)$ lower bound on the performance of a universal TSP algorithm where k is the number of terminals. Although [6] do not address universal Steiner tree problem directly, the $\Omega(k)$ lower bound for universal TSP implies an $\Omega(k)$ lower bound for universal Steiner tree as well, only when the algorithm returns spanning trees. However, this doesn't work for algorithms that return collections of vertex-to-root paths. Our result gives the first $\Omega(k)$ lower bound for the universal Steiner tree problem when the algorithm is allowed to return a collection of vertex-to-root paths.

Furthermore, even though our approach is somewhat similar, our proofs are simpler and the results are stronger in that we prove that the probability any randomized algorithm pays $o(\log n)$ times the optimum for a certain subset is exponentially small in the size of the client set. As explained earlier, these stronger lower bounds are crucial to our technique for proving privacy lower bounds. In particular, to our knowledge, no lower bounds for differentially private Steiner tree (even for weaker algorithms returning spanning trees instead of vertex-to-root paths) and TSP can be deduced from results of [6].

Organization. In Section 2, we establish an $\Omega(\log n)$ lower bound for the universal Steiner tree problem and the universal TSP. As mentioned above, the lower bound for the Steiner tree problem is for a more general class of algorithms which return a collection of paths instead of a single tree. The lower bound established are strong in the sense defined earlier, and thus give an $\Omega(\log n)$ lower bound for private Steiner tree as well as private TSP. We formalize the connection between strong lower bounds for universal problems and approximability of differentially private variants in Section 3. Finally, in interest of space, certain proofs have been omitted from the abstract and can be found in the full version of the paper [3].

2 Lower Bound Constructions

The metric spaces on which we obtain our lower bounds are shortest path metrics of expander graphs. Before exhibiting our constructions, we state a few known results regarding expanders that we use. An (n, d, β) expander is a d regular, n vertex graph with the second largest eigenvalue of its adjacency matrix $\beta < 1$. The girth g is the size of the smallest cycle and the diameter Δ is the maximum distance between two vertices. A t-step random walk on an expander picks a vertex uniformly at random, and at each step moves to a neighboring vertex uniformly at random.

Lemma 1. *[15] For any constant k, there exist (n, d, β) expanders, called Ramanujan graphs, with $d \geq k$, $\beta \leq \frac{2}{\sqrt{d}}$, girth $g = \Theta(\log n / \log d)$, and diameter $\Delta = \Theta(\log n / \log d)$.*

Lemma 2. *(Theorem 3.6, [10]) Given an (n, d, β) expander, and a subset of vertices B with $|B| = \alpha n$, the probability that a t-step random walk remains completely inside B is at most $(\alpha + \beta)^t$.*

Lemma 3. *(Follows from Theorem 3.10, [10]) Given an (n, d, β) expander, a subset of vertices B with $|B| = \alpha n$, and any $\gamma, 0 \leq \gamma \leq 1$, the probability that a t-step random walk visits more than γt vertices in B is at most $2^t \cdot (\alpha + \beta)^{\gamma t}$.*

2.1 Steiner Tree Problem

We consider a stronger class of algorithms that are allowed to return a distribution $\mathcal{D}$ on collections of paths $P := \{p_v : v \in V\}$, where each p_v is a path from v to the root r. As stated in the introduction, this class of algorithms captures as a special case algorithms that simply return a distribution on collection of spanning trees, since the latter induces a collection of paths. We prove the following theorem.

Theorem 1. *For any constant $\varepsilon > 0$ and for large enough n, there exists a metric space (V, c) on n vertices such that for any distribution $\mathcal{D}$ on collections of paths, there is a terminal set X of size $\Theta(\log n)$, such that*

$$\Pr_{P \leftarrow \mathcal{D}} \left[c(P[X]) = o\left(\frac{\log n}{1+\epsilon} \right) \mathtt{opt}_{ST}(X) \right] \leq \frac{1}{2} \exp(-\varepsilon |X|) \tag{1}$$

At a high-level, the idea underlying our proof is as follows. We choose as our underlying graph a Ramanujan graph G, and consider the shortest path metric induced by this graph. We show that for any fixed collection P of vertex-to-root paths, a terminal set generated by a random walk q of length $\Theta(\log n)$ in G has the following property with high probability: the edges on q frequently "deviate" from the paths in the collection P. These deviations can be mapped to cycles in G, and the high-girth property is then used to establish that the cost of the solution induced by P is $\Omega(\log n)$ times the optimal cost. Before proving Theorem 1, we establish the following corollaries of it.

Corollary 1. *(a) There is no $o(\log n)$-approximate universal Steiner tree algorithm. (b) There is no $o(k)$-approximate universal Steiner tree algorithm where k is the size of the terminal set. (c) For any $\varepsilon > 0$, there is no $o(\log n/(1+\varepsilon))$-approximate private algorithm with privacy parameter ε.*

Proof. The proofs of (a) and (b) are immediate by fixing ε to be any constant. The universal algorithm pays at least $\Omega(\log n)$ times the optimum with high probability, thus giving a lower bound of $\Omega(\log n)$ on the expected cost. To see (c), consider a differentially private algorithm $\mathcal{A}$ with privacy parameter ε. Let $\mathcal{D}$ be the distribution on the collection of paths returned by $\mathcal{A}$ when the terminal set is $\emptyset$. Let X be the subset of vertices corresponding to this distribution in Theorem 1. Let $\mathcal{P} := \{P : c(P[X]) = o(\frac{\log n}{1+\epsilon}) \cdot \mathtt{opt}_{ST}(X)\}$; we know $\Pr_{P \leftarrow \mathcal{D}}[P \in \mathcal{P}] \leq \frac{1}{2} \exp(-\varepsilon|X|)$. Let $\mathcal{D}'$ be the distribution on the collection of paths returned by $\mathcal{A}$ when the terminal set is X. By the definition of ε-differential privacy, we know that $\Pr_{P \leftarrow \mathcal{D}'}[P \in \mathcal{P}] \leq \exp(\varepsilon \cdot |X|) \cdot \left(\frac{1}{2}\exp(-\varepsilon|X|)\right) \leq 1/2$. Thus with probability at least $1/2$, the differentially private algorithm returns a collection of path of cost $\Omega\left(\frac{\log n}{1+\epsilon}\right) \cdot \mathtt{opt}_{ST}(X)$, implying the lower bound.

Note that the statement of Theorem 1 is much stronger than what is needed to prove the universal lower bounds. The proof of part (c) of the above corollary illustrates our observation that showing strong lower bounds for universal problems imply lower bounds for privacy problems. This holds more generally, and we explore this more in Section 3. We now prove Theorem 1.

Proof of Theorem 1: Consider an (n, d, β) expander as in Lemma 1 with degree $d \geq 2^{K(1+\epsilon)}$, where K is a large enough constant. The metric (V, c) is the shortest path metric induced by this expander. The root vertex r is an arbitrary vertex in V. We now demonstrate a distribution $\mathcal{D}'$ on terminal sets X such that $\varepsilon|X| \leq C_0 \log n$, for some constant C_0, and for any fixed collection of paths P,

$$\Pr_{X \leftarrow \mathcal{D}'} \left[c(P[X]) = o\left(\frac{\log n}{1+\varepsilon} \right) \mathtt{opt}_{ST}(X) \right] \leq \frac{1}{2} \exp(-C_0 \log n). \tag{2}$$

The lemma below is essentially similar to Yao's lemma [20] used for establishing lower bounds on the performance of randomized algorithms against oblivious adversaries; its proof is omitted.

Lemma 4. *Existence of a distribution $\mathcal{D}'$ satisfying* (2) *proves Thm 1.*

The distribution $\mathcal{D}'$ is defined as follows. Recall that the girth and the diameter of G are denoted by g and Δ respectively, and both are $\Theta\left(\frac{\log n}{\log d}\right)$. Consider a random walk q of t-steps in G, where $t = g/3$, and let X be the set of distinct vertices in the random walk. This defines the distribution on terminal sets. Note that each X in the distribution has size $|X| = O(\log n/\log d)$. We define C_0 later to be a constant independent of d, and thus since d is large enough, $\varepsilon|X| \leq C_0 \log n$.

Fix a collection of paths P. Since we use the shortest path metric of G, we may assume that P is a collection of paths in G as well. Let (v, v_1) be the first edge on the path p_v, and let $F := \{(v, v_1) : v \in V\}$ be the collection of all these first edges. The following is the crucial observation which gives us the lower bound. Call a walk $q = (u_1, \ldots, u_t)$ on t vertices *good* if at most $t/8$ of the edges of the form (u_i, u_{i+1}) are in F, and it contains at least $t/2$ distinct vertices.

We are now ready to complete the proof using the lemma below.

Lemma 5. *Let G be an (n, d, β) expander where d is a large constant ($\geq 2^{100}$, say) and $\beta = \frac{2}{\sqrt{d}}$. Suppose we mark an arbitrarily chosen subset of n edges in G as bad. Then the probability that a t step random walk contains at most $t/8$ bad edges and covers at least $t/2$ distinct vertices is at least $(1 - d^{-\Omega(t)})$.*

Lemma 6. *Let q be a good walk of length $t = g/3$ and let X be the set of distinct vertices in q. Then $c(P[X]) = \Omega(|X|g)$.*

Proof. Let X' be the vertices in X which do not traverse edges in F in the random walk q. Thus $|X'| \geq |X| - 2t/8 \geq |X|/2$. We now claim that $c(P[X']) \geq |X'|g/3$ which proves the lemma. For every $u \in X'$, let p'_u be the first $g/3$ edges in the path p_u (if p_u's length is smaller than $g/3$, $p'_u = p_u$). All the p'_u's are vertex disjoint: if p'_u and p'_v intersect then the union of the edges in p'_u, p'_v and the part of the walk q from v to u contains a cycle of length at most g contradicting that the girth of G is g. Thus, $c(P[X'])$, which is at least $c(\bigcup_{u \in X'} p'_u) \geq |X'|g/3 \geq |X|g/6$.

Call the set of edges F *bad*; note that the number of bad edges is at most n. Lemma 5 implies that the probability a t-step random walk is good is at least $(1 - d^{-\Omega(t)})$. Observe that this expression is $(1 - \exp(-C_0 \log n))$ for a constant C_0 independent of d. Furthermore, whenever q is a good walk, the set of distinct vertices X in q are at least $t/2$ in number; therefore $\texttt{opt}_{ST}(X) \leq t + \Delta = \Theta(|X|)$ since one can always connect X to r by travelling along q and then connecting to r. On the other hand, Lemma 6 implies that $c(P[X]) = \Omega(|X|g) = \Omega(\frac{\log n}{\log d}) \cdot \texttt{opt}_{ST}(X) = \Omega(\frac{\log n}{1+\varepsilon}) \cdot \texttt{opt}_{ST}(X)$, by our choice of d. This gives that

$$\Pr_{X \leftarrow \mathcal{D}'}[c(P[X]) \leq o\left(\frac{\log n}{1+\varepsilon}\right) \texttt{opt}_{ST}(X)] \leq \frac{1}{2}\exp(-C_0 \log n)$$

where C_0 is independent of d. Thus, $\mathcal{D}'$ satisfies (2), implying, by Lemma 4, Theorem 1. □

2.2 Traveling Salesman Problem

We now show an $\Omega(\log n)$ lower bound for the traveling salesman problem. In contrast to our result for the Steiner tree problem, the TSP result is slightly weaker result in that it precludes the existence of $o(\log n)$-approximate private algorithms for arbitrarily small constant privacy parameters only.

We remark here that a lower bound for universal TSP implies a similar lower bound for any universal Steiner tree algorithm which returns a distribution on spanning trees. However, this is not the case when the algorithm returns a collection of paths; in particular, our next theorem below does not imply Theorem 1 even in a weak sense, that is, even if we restrict the parameter ε to be less than the constant ε_0.

Theorem 2. *There exists a metric space (V, c) and a constant ε_0, such that for any distribution $\mathcal{D}$ on tours σ of V, there exists a set $X \subseteq V$ of size $\Theta(\log n)$ such that*

$$\Pr_{\sigma \leftarrow \mathcal{D}}[c(\sigma_X) = o(\log n) \cdot \mathtt{opt}_{TSP}(X)] \leq \frac{1}{2}\exp(-\varepsilon_0|X|)$$

At a high level, the idea as before is to choose as our underlying graph a Ramanujan graph G, and consider the shortest path metric induced by this graph. We show that for any fixed permutation σ of vertices, with high probability a *pair* of random walks, say q_1, q_2, has the property that they frequently alternate with respect to σ. Moreover, with high probability, every vertex on q_1 is $\Omega(\log n)$ distance from every vertex in q_2. The alternation along with large pairwise distance between vertices of q_1 and q_2 implies that on input set defined by vertices of q_1 and q_2, the cost of the tour induced by σ is $\Omega(\log n)$ times the optimal cost.

As stated in the Introduction, Gorodezky et al. [6] also consider the shortest path metric on Ramanujan expanders to prove their lower bound on universal TSP. However, instead of taking clients from two independent random walks, they use a single random walk to obtain their set of 'bad' vertices. Seemingly, our use of two random walks makes the proof easier, and allows us to make a stronger statement: the RHS in the probability claim in Theorem 2 is exponentially small in $|X|$, while [6] implies only a constant. This is not sufficient for part (c) of the following corollary.

As in the case of Steiner tree problem, we get the following corollaries of the above theorem.

Corollary 2. *(a) There is no $o(\log n)$-approximate universal TSP algorithm. (b) There is no $o(k)$-approximate universal TSP algorithm where k is the size of the terminal set. (c) There exists $\varepsilon_0 > 0$ such that there is no $o(\log n)$-approximate private algorithm with privacy parameter at most ε_0.*

3 Strong Universal Lower Bounds Imply Privacy Lower Bounds

Suppose Π is a minimization problem whose instances are indexed as tuples (I, X). The first component I represents the part of the input that is accessible to the algorithm (and is public); for instance, in the Steiner tree and the TSP example, this is the metric space (V, c) along with the identity of the root. The second component X is the part of the input which is either unknown beforehand, or corresponds to the private input. We assume that X is a subset of some finite universe $U = U(I)$. In the Steiner tree and TSP example, X is the set of terminals which is a subset of all the vertices. An instance (I, X) has a set of feasible solutions $\mathcal{S}(I, X)$, or simply $\mathcal{S}(X)$ when I is clear from context, and let $\mathcal{S} := \bigcup_{X \subseteq U} \mathcal{S}(X)$. In the case of Steiner trees, $\mathcal{S}(X)$ is the collection of rooted trees containing X; in the case of TSP it is the set of tours spanning $X \cup r$. Every solution $S \in \mathcal{S}$ has an associated cost $c(S)$, and $\mathtt{opt}(X)$ denotes the solution of minimum cost in $\mathcal{S}(X)$.

We assume that the solutions to instances of Π have the following *projection* property. Given any solution $S \in \mathcal{S}(X)$ and any $X' \subseteq X$, S induces a unique solution in $\mathcal{S}(X')$, denoted by $\pi_{X'}(S)$. For instance, in case of the Steiner tree problem, a rooted tree spanning vertices of X maps to the unique minimal rooted tree spanning X'. Similarly, in the TSP, an ordering of vertices in X maps to the induced ordering of X'. In this framework, we now define approximate universal and differentially private algorithms.

An *α-approximate universal algorithm* for Π takes input I and returns a distribution $\mathcal{D}$ over solutions in $\mathcal{S}(U)$ with the property that for any $X \subseteq U$, $\mathbf{E}_{S \leftarrow \mathcal{D}}[c(\pi_X(S))] \leq \alpha \cdot \mathtt{opt}(I, X)$. An *$\alpha$-approximate differentially private algorithm* with *privacy parameter ε* for Π takes as input (I, X) and returns a distribution $\mathcal{D}_X$ over solutions in $\bigcup_{Y \supseteq X} \mathcal{S}(Y)$ that satisfies the following two properties. First, for all (I, X), $\mathbf{E}_{S \leftarrow \mathcal{D}_X}[c(\pi_X(S))] \leq \alpha \cdot \mathtt{opt}(I, X)$. Second, for any set of solutions $\mathcal{F}$ and for any pair of sets X and X' with symmetric difference exactly 1, we have

$$\exp(-\varepsilon) \cdot \Pr_{S \leftarrow \mathcal{D}_{X'}}[S \in \mathcal{F}] \leq \Pr_{S \leftarrow \mathcal{D}_X}[S \in \mathcal{F}] \leq \exp(\varepsilon) \cdot \Pr_{S \leftarrow \mathcal{D}_{X'}}[S \in \mathcal{F}]$$

It is easy to see that any α-approximate universal algorithm is also an α-approximate differentially private algorithm with privacy parameter $\varepsilon = 0$; the distribution $\mathcal{D}_X := \mathcal{D}$ for every X suffices. We now show a converse relation: lower bounds for universal algorithms with a certain additional property imply lower bounds for private algorithms as well. We make this precise.

Fix $\rho : [n] \to [0, 1]$ to be a non-increasing function. We say that an (α, ρ) *lower bound* holds for universal algorithms if there exists I with the following property. Given any distribution $\mathcal{D}$ on $\mathcal{S}(U)$, there exists a subset $X \subseteq U$ such that

$$\Pr_{S \leftarrow \mathcal{D}}[c(\pi_X(S)) \leq \alpha \cdot \mathtt{opt}(I, X)] \;\leq\; \rho(|X|) \tag{3}$$

We say that the set X achieves the (α, ρ) lower bound. It is not hard to see that when ρ is a constant function bounded away from 1, an (α, ρ) lower bound is equivalent to an $\Omega(\alpha)$ lower bound on universal algorithms.

Theorem 3. *Suppose there exists a (α, ρ) lower bound for universal algorithms for a problem Π. Then any ε-private algorithm for Π with $\varepsilon \leq \varepsilon_0 := \inf_X \frac{1}{|X|} \ln\left(\frac{1}{2\rho(|X|)}\right)$ has an approximation factor of $\Omega(\alpha)$.*

Proof. Let I be an instance that induces the (α, ρ) lower bound. Consider the output of a differentially private algorithm $\mathcal{A}$ with privacy parameter $\varepsilon < \varepsilon_0$, on the input pair $(I, \emptyset)$. Let $\mathcal{D}$ be the distribution on the solution set $\mathcal{S}$. We first claim that all S in the support of $\mathcal{D}$ lie in $\mathcal{S}(U)$. Suppose not and suppose there is a solution $S \in \mathcal{S}(Z) \setminus \mathcal{S}(U)$, for some $Z \subset U$, which is returned with non-zero probability. By the definition of differential privacy, this solution must be returned with non-zero probability when $\mathcal{A}$ is run with (I, U), contradicting feasibility since $S \notin \mathcal{S}(U)$.

Thus, $\mathcal{D}$ can be treated as a universal solution for Π. Let X be the set which achieves the (α, ρ) lower bound for $\mathcal{D}$, and let $\mathcal{F} := \{S \in \mathcal{S}(X) : c(S) \leq \alpha \cdot \mathtt{opt}(I, X)\}$. By the definition of the lower bound, we know that $\Pr_{S \leftarrow \mathcal{D}}[S \in \mathcal{F}] \leq \rho(|X|)$. Let $\mathcal{D}'$ be the output of the algorithm $\mathcal{A}$ when the input is (I, X). By definition of differential privacy, $\Pr_{S \leftarrow \mathcal{D}'}[S \in \mathcal{F}] \leq \exp(\varepsilon \cdot |X|) \cdot \rho(|X|) \leq 1/2$, from the choice of ε. This shows a lower bound on the approximation factor of any differential private algorithm for Π with parameter $\varepsilon < \varepsilon_0$.

Acknowledgements. We would like to thank Kunal Talwar for posing the question on differentially private Steiner trees at Bellairs workshop on approximation algorithms which started all this work, Bruce Shepherd, Adrian Vetta, and Mohit Singh for inviting DC to the workshop, and Anupam Gupta for his valuable comments on a previous draft.

References

1. Archer, A.: Two $O(\log^* k)$-approximation algorithms for the asymmetric k-center problem. In: Proceedings, MPS Conference on Integer Programming and Combinatorial Optimization (IPCO), pp. 1–14 (2010)
2. Bartal, Y.: On approximating arbitrary metrices by tree metrics. In: ACM Symp. on Theory of Computing (STOC), pp. 161–168 (1998)
3. Bhalgat, A., Chakrabarty, D., Khanna, S.: Optimal lower bounds for universal and differentially private steiner trees and tsps. Technical report, `http://arxiv.org/abs/1011.3770`
4. Dwork, C.: Differential privacy. In: Proceedings, International Colloquium on Automata, Languages and Processing, pp. 1–12 (2006)
5. Fakcharoenphol, J., Rao, S., Talwar, K.: A tight bound on approximating arbitrary metrics by tree metrics. In: ACM Symp. on Theory of Computing (STOC), pp. 448–455 (2003)

6. Gorodezky, I., Kleinberg, R.D., Shmoys, D.B., Spencer, G.: Improved lower bounds for the universal and a priori tsp. In: Proceedings, International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, pp. 178–191 (2010)
7. Gupta, A., Hajiaghayi, M., Räcke, H.: Oblivious network design. In: Proceedings, ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 970–979 (2006)
8. Gupta, A., Ligett, K., McSherry, F., Roth, A., Talwar, K.: Differentially private approximation algorithms. In: Proceedings, ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 1106–1125 (2010)
9. Hajiaghayi, M., Kleinberg, R., Leighton, F.T.: Improved lower and upper bounds for universal tsp in planar metrics. In: Proceedings, ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 649–658 (2006)
10. Hoory, S., Linial, N., Wigderson, A.: Expander graphs and their applications. Bull. of the Amer. Soc. 43(4), 439–561 (2006)
11. Imase, M., Waxman, B.M.: Dynamic steiner tree problem. SIAM J. Discrete Math. 4(3), 369–384 (1991)
12. Jia, L., Lin, G., Noubir, G., Rajaraman, R., Sundaram, R.: Universal approximations for tsp, steiner tree, and set cover. In: ACM Symp. on Theory of Computing (STOC), pp. 386–395 (2005)
13. Leighton, F.T., Rao, S.: An approximate max-flow min-cut theorem for uniform multicommodity flow problems with application to approximation algorithms. In: Proceedings, IEEE Symposium on Foundations of Computer Science (FOCS), pp. 422–431 (1988)
14. Linial, N., London, E., Rabinovich, Y.: The geometry of graphs and some of its algorithmic applications. Combinatorica 15(2), 215–246 (1995)
15. Lubotzky, A., Phillips, R., Sarnak, P.: Ramanujan graphs. Combinatorica 4, 261–277 (1988)
16. Moitra, A.: Approximation algorithms for multicommodity-type problems with guarantees independent of the graph size. In: Proceedings, IEEE Symposium on Foundations of Computer Science (FOCS), pp. 3–12 (2009)
17. Moitra, A., Leighton, F.T.: Extensions and limits to vertex sparsification. In: ACM Symp. on Theory of Computing (STOC), pp. 47–56 (2010)
18. Panigrahy, R., Vishwanathan, S.: An $O(\log^* n)$ approximation algorithm for the asymmetric p-center problem. J. Algorithms 27(2), 259–268 (1998)
19. Talwar, K.: Problem 1. In: Open Problem in Bellairs Workshop on Approximation Algorithms, Barbados (2010), `http://www.math.mcgill.ca/~vetta/Workshop/openproblems2.pdf`
20. Yao, A.C.-C.: Probabilistic computations: Towards a unified measure of complexity. In: Proceedings, IEEE Symposium on Foundations of Computer Science (FOCS), pp. 222–227 (1977)