Approximating TSP on Metrics with Bounded Global Growth*

T-H. Hubert Chan   Anupam Gupta

Abstract

The Traveling Salesman Problem (TSP) is a canonical NP-complete problem which is known to be MAX-SNP hard even on (high-dimensional) Euclidean metrics [39]. In order to circumvent this hardness, researchers have been developing approximation schemes for low-dimensional metrics [4, 38] (under different notions of dimension). However, a feature of most current notions of metric dimension is that they are "local": the definitions require *every* local neighborhood to be well-behaved. In this paper, we consider the case *when the metric is less restricted: it has a few "dense" regions, but is "well-behaved on the average"?*

To this end, we define a global notion of dimension which we call the *correlation dimension* (denoted by $\dim_C$), which generalizes the popular notion of doubling dimension. In fact, the class of metrics with $\dim_C = O(1)$ not only contains all doubling metrics, but also contains some metrics containing uniform submetrics of size $\sqrt{n}$. We first show, using a somewhat "local" argument, that one can solve TSP on these metrics in time $2^{O(\sqrt{n})}$; we then take advantage of the global nature of TSP (and the global nature of our definition) to give a $(1 + \varepsilon)$-approximation algorithm that runs in *sub-exponential time*: i.e., in $2^{O(n^{\delta}\varepsilon^{-4\dim_C})}$-time for every constant $0 < \delta < 1$.

1 Introduction

Distance functions are ubiquitous, arising as distances from home to work, round-trip delays between hosts on the Internet, dissimilarity measures between documents, and many other applications. As a simplifying assumption, theoreticians often assume that the distance function in question satisfies the triangle inequality and hence is a metric.

However, some problems remain hard even when the underlying distance function is a metric, an example of which is the Traveling Salesman Problem (TSP). Papadimitriou and Yannakakis [34] showed that TSP is MAX-SNP hard in general for metrics whose distances are either 1 or 2. Indeed, even for more structured metrics such as Euclidean metrics, Trevisan [39] showed that the problem remains MAX-SNP hard if the Euclidean dimension is unbounded. On the other hand, Arora [4] gave a PTAS for TSP on low dimensional Euclidean metrics. A natural and basic question that arises in the study of metric spaces is: *How do we quantify the complexity of metric spaces?* More specifically, *which classes of metric spaces admit efficient algorithms for TSP?*

It is not surprising that metrics induced by special classes of graphs admit efficient TSP algorithms. For instance, for graphs with bounded treewidth, Arnborg and Proskurowski [3] gave a dynamic program that solves TSP on the induced metrics exactly in linear time. For metrics induced by weighted planar graphs, the best known algorithm is by Klein [23], who gave a $(1+\varepsilon)$-approximation algorithm that runs in linear time $O(c^{1/\varepsilon^2} n)$, where $c > 0$ is some constant. Grigni [17] gave QPTAS's for metrics induced by minor-forbidding graphs and bounded-genus graphs.

An orthogonal line of recent work sought to extend the notion of dimension in geometric spaces to capture the complexity of arbitrary metric spaces. One of the more popular concepts has been that of *doubling dimension* $\dim_D(M)$ of a metric M [18], which generalizes the notion of Euclidean dimension, i.e., $\dim_D(\mathbb{R}^d, \ell_p) = \Theta(d)$. Doubling dimension has proved to be fairly useful: over the past three years, several algorithms have since been developed whose performance (run-time, space) can be given by functions $F(|V|, \dim_D(M))$, which give better quantification than those obtained for general metrics. For instance, Talwar [38] gave a $(1 + \varepsilon)$-approximation algorithm for TSP such that for metrics with doubling dimension $\dim_D(M)$ at most k, the algorithm runs in time $2^{(\frac{k}{\varepsilon} \log n)^{O(k)}}$.

Despite its popularity, doubling dimension has some drawbacks: perhaps the biggest one is being that a space with low $\dim_D$ cannot have "large dense clusters".[1]

---

[1] More precisely, the doubling dimension is defined so that any set that is almost equilateral in a metric of dimension $\dim_D$ can only have $2^{\dim_D}$ points in it; the precise definition of doubling appears in Section 2.

"Head": $\sqrt{n}$-clique

"Tail": path with $n - \sqrt{n}$ nodes

(a) Lollipop

$\sqrt{n}$-sized "hard" instance of $(1,2)$-TSP

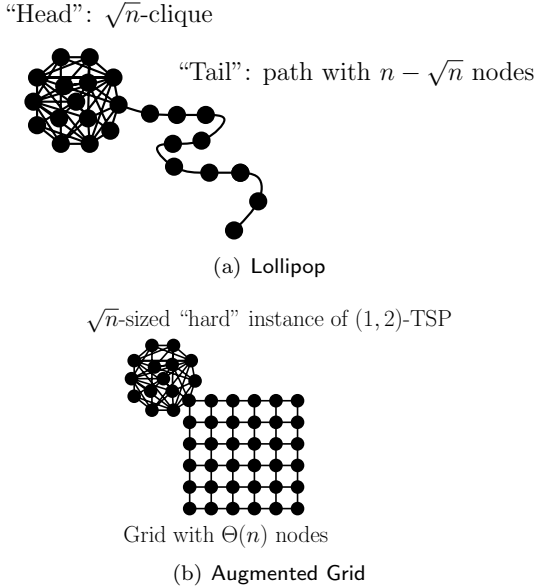Grid with $\Theta(n)$ nodes

(b) Augmented Grid

Figure 1: Very simple examples of metrics with low correlation dimension.

This strict definition makes it difficult to use it to model real networks, which tend to be well-behaved "on the average", but often have a few regions of "high density". We define a new notion of dimension, the **correlation dimension** which captures the idea of being *"low-dimensional on average"*. We give structural results as well as algorithms for spanners and TSP for metrics with low correlation dimension. Our definitions are inspired by work on the *correlation fractal dimension* in physics [16] and in databases [7].

Our Results and Techniques. Given a finite metric $M = (V, d)$, let $\mathbf{B}(x, r)$ denote the ball around u of radius r. The *correlation dimension* is defined as the smallest constant k such that

$$(1.1) \qquad \sum_{x \in V} |\mathbf{B}(x, 2r)| \leq 2^k \cdot \sum_{x \in V} |\mathbf{B}(x, r)|,$$

and moreover, this inequality must hold under taking any net of the metric M. (A more formal definition is given in Section 2.) Note that this definition is an "average" version of the bounded-growth rate used by [35, 22], and hence should be more general than that notion. We show that in fact, correlation dimension is *even more general than doubling dimension*:

THEOREM 1.1. (CORRELATION GENERALIZES DOUBLING) *Given a metric M, the correlation dimension is bounded above by a constant times the doubling dimension.*

Moreover, correlation dimension is strictly more general

than doubling dimension: adding a clique of size $O(\sqrt{n})$ to a doubling metric does not change its correlation dimension by much, but completely destroys its doubling dimension. (Some examples are given in Figure 1. One can be convinced that each of these example metrics has "low complexity on average," which is precisely what correlation dimension tries to capture.)

The following theorems show the algorithmic potential of this definition.

THEOREM 1.2. (EMBEDDING INTO SMALL TREEWIDTH GRAPHS) *Given any constant $0 < \varepsilon < 1$ and k, metrics with correlation dimension at most k can be embedded into a distribution of graphs with treewidth $\tilde{O}_{k,\varepsilon}(\sqrt{n})$ and distortion $1 + \varepsilon$.*

This immediately allows us to get $2^{\tilde{O}(\sqrt{n})}$-time algorithms for all problems that can be solved efficiently on small-treewidth graphs, including the traveling salesman problem. Moreover, Theorem 1.2 is tight, since metrics with bounded $\dim_C$ can contain $O(\sqrt{n})$-sized cliques.

However, we can do much better for the TSP despite the presence of these $O(\sqrt{n})$-sized cliques (or other complicated metrics of that size); we can make use of the global nature of the TSP problem (and the corresponding global nature of $\dim_C$) to get the following result.

THEOREM 1.3. (APPROXIMATION SCHEMES FOR TSP) *Given any metric M with $\dim_C(M) = k$, the TSP can be solved to within an expected $(1 + \varepsilon)$-factor in time $2^{O(n^\delta \varepsilon^{-k})}$ for any constant $\delta > 0$.*

Hence, given constants ε, k, the algorithm runs in *sub-exponential* time. (Recall that sub-exponential time is $\cap_{\delta > 0} \mathsf{DTIME}(2^{n^\delta})$.) As we will see later, the best exponent in the expression above that we can show is $(\varepsilon^{-1} 2^{\sqrt{\log n \log \log n}})^{4k}$.

We show that while metrics with bounded correlation dimension cannot in general have $(1 + \varepsilon)$-stretch spanners with a linear number of edges, we can get some improvement over general metrics.

THEOREM 1.4. (SPARSE SPANNERS) *Given any $0 < \varepsilon < 1$, any metric with correlation dimension k has a spanner with $O(n^{3/2} \varepsilon^{-O(k)})$ edges and stretch $(1 + \varepsilon)$. Moreover, there exist metrics with $\dim_C = 2$ and for each of which any 1.5-stretch spanner has $\Omega(n^{3/2})$ edges.*

1.1 Related Work Many notions of dimension for metric spaces (and for arbitrary measures) have been proposed; see the survey by Clarkson [10] for the definitions, and for their applicability to near-neighbor (NN) search. Some of these give us strong algorithmic

properties which are useful beyond NN-searching. For instance, the *low-growth rate* of a metric space requires that for all $x \in V$ and all r, $|\mathbf{B}(x, 2r)|$ is comparable to $|\mathbf{B}(x, r)|$. This was used in [35, 22, 21] to develop algorithms for object location in general metrics, and in [25, 2], for routing problems.

A large number of algorithms have been developed for doubling metrics; we only mention a few of them here. For example, the doubling dimension has been considered in the context of near-neighbor searching [29, 30, 8, 19, 11, 10], for the TSP and other optimization problems [38], for low-stretch compact routing [38, 9, 37, 1, 27, 28], for sparse spanners [9, 19], and for many other applications [24, 31].

For Euclidean metrics, the first approximation schemes for TSP and other problems were given by Arora [4] and Mitchell [32]. See, for example [12, 5, 13, 26], for subsequent algorithms. The runtime of Arora's algorithm [4] was $O(n(\log n)^{O(\sqrt{k} \cdot \frac{1}{\varepsilon})^{k-1}})$, which was improved to $2^{(\frac{k}{\varepsilon})^{O(k)}} n + O(kn \log n)$ [36]. For $(1 + \varepsilon)$-approximation for TSP on doubling metrics, the best known running time is $2^{(\frac{k}{\varepsilon} \log n)^{O(k)}}$ [38]. Here, the parameter k is the doubling dimension or the Euclidean dimension in the corresponding cases.

1.2 Earlier Notions of Correlation Dimension

The concept of correlation fractal dimension [16] was used by physicists to distinguish between a chaotic source and a random source; while it is closely related to other notions of fractal dimension, it has the advantage of being easily computable. Let us define it here, since it may be useful to compare our definitions with the intuition behind the original definitions.

Consider an infinite set V. If $\sigma = \{x_i\}_{i \geq 1}$ is a sequence of points in V, the *correlation sum* is defined as $C_n(r) = \frac{1}{n^2} |\{(i, j) \in [n] \times [n] \mid d(x_i, x_j) \leq r\}|$ (i.e., the fraction of pairs at distance at most r from each other). The *correlation integral* is then $C(r) = \lim_{n \to \infty} C_n(r)$, and the *correlation fractal dimension* for σ is defined to be $\lim_{r \to 0} \lim_{\epsilon \to 0} \frac{\log C((1+\epsilon)r) - \log C(r)}{\log(1+\epsilon)}$. Hence, given a set of points, the correlation fractal dimension quantifies the rate of growth in the number of points which can see each other as their range-of-sight increases. In the next section, we will define a version of this definition for finite sets.

These notions have been used before in algorithmic contexts: the correlation fractal dimension was used by Belussi and Faloutsos [7, 33] for estimating the selectivity of spatial queries; Faloutsos and Kamel [15] also used fractal dimension to analyze R-trees.

2 Correlation Dimension: Definition and Motivation

Given a finite metric $M = (V, d)$, we denote the number of points $|V|$ by n. For radius $r > 0$, we define the ball $\mathbf{B}(x, r) = \{y \in V \mid d(x, y) \leq r\}$. Given $U \subseteq V$, define $\mathbf{B}_U(x, r) = \mathbf{B}(x, r) \cap U$. Recall that a subset $N \subseteq V$ is an ε-*cover* for V if for all points $x \in V$, there is a covering point $y \in N$ with $d(x, y) \leq \varepsilon$. A subset $N \subseteq V$ is an ε-*packing* if for all $x, y \in N$ such that $x \neq y$, $d(x, y) > \varepsilon$. A subset $N \subseteq V$ is an ε-*net* if it is both an ε-cover and an ε-packing. A set $N \subseteq V$ is a *net* if it is an ε-net for some ε.

Inspired by the definitions mentioned in Section 1.2, we give the following definition:

DEFINITION 2.1. (CORRELATION DIMENSION) *A metric $M = (V, d)$ has* correlation dimension $\dim_C(M)$ *at most k if for all $r > 0$, the inequality*

$$(2.2) \qquad \sum_{x \in N} |\mathbf{B}_N(x, 2r)| \leq 2^k \cdot \sum_{x \in N} |\mathbf{B}_N(x, r)|$$

holds for all nets $N \subseteq V$.

In other words, we want to ensure that the average growth rate of the metric M is not too large, and the same holds for any net N of the metric. Recall that the *doubling dimension* $\dim_D(M)$ is the least k such that every ball $\mathbf{B}(x, r)$ of radius r can be covered by at most 2^k balls of radius $r/2$ [18]. The *strong doubling dimension*[2] is the least k such that

$$(2.3) \qquad |\mathbf{B}(x, 2r)| \leq 2^k |\mathbf{B}(x, r)|$$

for all $x \in V$ and radius r. We know that the strong doubling dimension is no more than $4 \dim_D$ [18]. It follows directly from the definition (2.3) that the correlation dimension is no more than the strong doubling dimension; more surprisingly, the following result is true as well. We give its proof in the full version.

THEOREM 2.1. *For any metric space M, $\dim_C(M) \leq O(\dim_D(M))$.*

Hence the *class of bounded correlation dimension metrics contains the class of doubling metrics.* The converse is not true: metrics with bounded $\dim_C$ can be much richer. Consider, for instance, the unweighted 2-d grid with $\dim_D = \dim_C = O(1)$. Now attaching an unweighted clique (or, say, a metric with all distances between 1 and 2) on $O(\sqrt{n})$ vertices to one of the vertices of the grid: one can verify that the induced metric still has $\dim_C = O(1)$, but the $\dim_D$ jumps to $\frac{1}{2} \log n$.

The reader wondering about why the bounded average growth property (2.2) is required to hold for

---

[2]This quantity has been described as the KR-dimension in [18]; we use this name due to [8] to keep matters simple.

every net of M in Definition 2.1 is referred to the full version for detailed discussion: loosely, the definition becomes too inclusive without this restriction.

A very useful property of correlation dimension is that it still has "small" nets. (Of course, since we allow large cliques, they cannot be as small as for doubling dimension):

LEMMA 2.1. (SMALL NETS) *Consider a metric* $M = (V, d)$ *with* $\dim_C(M) \leq k$. *Suppose S is an R-packing with diameter D. If we add more points to S and obtain an R-net N for (V, d), then the size of the packing satisfies* $|S| \leq (2D/R)^{k/2} \cdot \sqrt{|N|}$.

Proof. Observe that $|S|^2 \leq \sum_{x \in N} |\mathbf{B}_N(x, D)|$. By applying the definition of correlation dimension repeatedly, we have for each integer $t \geq 0$,

$$(2.4) \quad \sum_{x \in N} |\mathbf{B}_N(x, D)| \leq 2^{kt} \sum_{x \in N} |\mathbf{B}_N(x, D/2^t)|.$$

Setting $t = \lceil \log_2(D/R) \rceil$ gives the required result. $\square$

Hence, given any metric with $\dim_C = O(1)$, any near-uniform set in the metric has size at most $O(\sqrt{n})$, and hence λ, the doubling constant [18] of this metric is also $O(\sqrt{n})$.

At this point, it is worthwhile to mention that because property (2.2) is required to hold for *every net* of M in Definition 2.1, it is hard to approximate the correlation dimension of a given metric.

THEOREM 2.2. *Given a metric* $M = (V, d)$ *with n points, it is NP-hard to distinguish between the cases* $\dim_C(M) = O(1)$ *and* $\dim_C(M) = \Omega(\log n)$.

The proof of Theorem 2.2 involves a reduction from the MAXIMUM INDEPENDENT SET [20] problem, and is given in the full version. Observe that this result rules out any non-trivial approximation of the correlation dimension; however, this does not necessarily rule out using correlation dimension for the design of algorithms. In particular, the algorithms we design do not require us to know the correlation dimension of the input metric up-front; while the TSP approximation algorithm of Section 5 seems to require this information at first glance, this issue can be resolved using standard "guess-and-double" ideas.

3 Sparse Spanners

We begin our study of metrics with small correlation dimension with a simple construction of sparse spanners; this will also serve to introduce the reader to some of the basic concepts we will use later. In this section, we show that metrics with bounded correlation dimension admit $(1+\varepsilon)$-stretch spanners with $O_\varepsilon(\min\{n^{1.5}, n \log \Delta\})$

edges, where $\Delta = \frac{\max_{x,y} d(x,y)}{\min_{x,y} d(x,y)}$ is the *aspect ratio* of the metric. This should be contrasted with a trivial lower bound for general metrics: any spanner with stretch less than 3 for $K_{n,n}$ requires $\Omega(n^2)$ edges.

THEOREM 3.1. (SPARSE SPANNER THEOREM) *Given a metric* $M = (V, d)$ *with* $\dim_C(M) \leq k$, *and* $\varepsilon > 0$, *there exists a* $(1 + \varepsilon)$-*spanner with* $\varepsilon^{-O(k)} \min\{n^{1.5}, n \log \Delta\}$ *edges.*

The algorithm for constructing sparse spanners for metrics with bounded correlation dimension is the same as that for doubling metrics in [9]; the proofs, of course, are different, and are deferred to the full version.

Note that for metrics with bounded doubling dimension, one can get a $(1+\varepsilon)$-spanners with $O(n\varepsilon^{-O(k)})$ edges [9, 19]. However, in the full version of the paper, we show that such a result is not possible with bounded correlation dimension, and that the upper bound in Theorem 3.1 is indeed tight.

THEOREM 3.2. (LOWER BOUND ON SPARSITY) *There exists a family of metrics with bounded correlation dimension such that for each metric, any* 1.5-*stretch spanner has at least* $\Omega(n^{1.5})$ *edges.*

4 Algorithms for Metrics with Bounded Correlation Dimension

Having defined the notion of correlation dimension, and having seen a simple warm-up (obtaining sparse spanners), we now turn to devising algorithms for metric spaces, whose performance is parameterized by the correlation dimension of the underlying metric space. This task is complicated by two issues:

(1) Global versus Local Properties. The notion of correlation dimension is *global*, in the sense that while there may be pockets of "high-complexity" in a metric with low $\dim_C$, the complexity is "low on the average". One should compare this to previous notions of dimension like doubling, where the metric was well-structured in every region and at every scale, and thus local arguments would usually suffice to give good algorithms. In sharp contrast, *we are forced to develop algorithms that take into account this global averaging.*

As an example, consider the TSP: suppose the input graph consists of a max-SNP hard $(1, 2)$-TSP instance on $\sqrt{n}$ nodes, which is attached to one vertex of a unit grid. If we want to obtain a $(1 + \varepsilon)$ approximation to TSP, our algorithm would have to cluster the graph into the "easy" part (the grid), and the "complicated" part (the $(1, 2)$-TSP instance), and perhaps run a (Q)PTAS on the former part and a constant approximation algorithm on the latter part. Of course, the input metric

with $\dim_C = O(1)$ may not have such an obvious clustering.

(2) Doubling results may not be directly applicable. As noted in the discussion after Lemma 2.1, metrics with $\dim_C = O(1)$ cannot have near-uniform sets of size $\omega(\sqrt{n})$, and hence their doubling dimension is at most $\frac{1}{2}\log_2 n + O(1)$. Hence, while we can conceivably use results for doubling metrics, most of the current results are no longer interesting for such a high value of doubling dimension : e.g., the result for TSP on doubling metrics has a running time of $\exp\{(\varepsilon^{-1}\log n)^{O(\dim_D)}\}$, and hence plugging in $\dim_D = \frac{1}{2}\log_2 n$ does worse than $2^{O(n)}$, the running time for an exact algorithm. Again, our algorithms will try to avoid this simple-minded reduction to doubling, even though they will rely on many ideas developed in the doubling metrics literature.

In the rest of the paper, the two main algorithmic results we present are:

(1) Weak TSP Approximation & Embedding into Small Treewidth Graphs. We first show how to solve the TSP on metrics with low correlation dimension within $(1 + \varepsilon)$ in time $2^{\sqrt{n}\cdot(\varepsilon^{-1}\log n)^{O(\dim_C)}}$. As a by-product, we also get Theorem 1.2: a random embedding of the original metric into a graph with treewidth $\sqrt{n}\cdot(\varepsilon^{-1}\log n)^{O(\dim_C)}$. Details of this result appear in Section 4.1. To prove these results, we adapt the ideas of Arora [4] and Talwar [38]. Observe that all the previous proofs use "$O(1)$-padded decompositions," and metrics with small $\dim_C$ may not admit such good padded decompositions, since padding is a local property, and our metric may have some dense regions. We show how to get around this requirement: we use known padded decompositions with poorer padding guarantees. Moreover, we will see that we also need to carefully alter the boundaries of clusters to serve our purpose.

(2) $(1 + \varepsilon)$-Approximations in Sub-exponential Time. The ideas we use for the previous algorithm are still fairly local, and hence do not fully use the power of having small correlation dimension. In Section 5, we show how to improve our partitioning scheme, and use an improved global charging scheme to get our main result Theorem 1.3: an approximation scheme for TSP that runs in sub-exponential time.

4.1 An Algorithm for TSP in Time $2^{\tilde{O}(\sqrt{n})}$ Given an $\varepsilon \leq 1$, we consider randomized $(1+\varepsilon)$-approximation algorithms for TSP on a metric $M = (V, d)$ on n points and $\dim_C = k$. Let OPT be the cost of the optimal TSP. As is well-known, we can assume the aspect ratio is n/ε (see, e.g., [4, 38]), Moreover, we assume that $\varepsilon > 1/n$,

or else we can solve it exactly in $2^{O(\varepsilon^{-1}\log \varepsilon^{-1})}$-time.

We use the following main ideas, which were also used in obtaining known (Q)PTAS's for TSP [4, 38]:
(a) We find a good probabilistic hierarchical decomposition into clusters with geometrically decreasing diameters, **(b)** we choose small set of *portals* in each cluster in this decomposition by taking a suitably fine net of the cluster, and force the tour to enter and leave the cluster using only these portals, i.e., the tour is *portal-respecting*. The main structure lemma shows that the expected cost of the best portal-respecting tour is at most $(1 + \varepsilon)$ times its original cost. Finally, **(c)** we find the best portal respecting tour using dynamic programming in a way similar to those used by Arnbourg and Proskurowski [3] and Arora [4]. For a cluster C, if there are only B portals among all its child clusters, the time to build the table for C is at most $B^{O(B)} = 2^{O(B \log B)}$. Since the total number of clusters is poly(n), total runtime is poly$(n)2^{O(B \log B)}$. Note that for doubling metrics, since each cluster had only $2^{O(\dim_D)}$ child clusters, each with $O(\varepsilon^{-1}\log n)^{O(\dim_D)}$ portals, the runtime is quasi-polynomial [38].

The main technical problem we face is that while we can ensure the number of portals in any single cluster are at most $\approx O(\sqrt{n})$ using Lemma 2.1, each cluster may have as many as $\sqrt{n}$ child clusters, and hence the size B of the union of portals for all the child clusters may be close to $\Theta(n)$. To take care of this problem, we need to find a new partitioning and portaling scheme, such that the union of the portals in each cluster *and in all its child clusters* has size only $\tilde{O}(\sqrt{n})$; clearly this will require us to do the partitioning and portal-creation steps in a dependent fashion, with each step guiding the other.

4.2 A Partitioning and Portaling Algorithm In this section we give a concrete construction of a probabilistic hierarchical decomposition and portaling scheme such that both the padding parameter and the number of child portals for each cluster are small. Observe that if the child portals of each cluster form a packing, then using the bounded correlation dimension assumption and Lemma 2.1, we can show that B is small for each cluster. If we use a standard hierarchical decomposition (e.g., the one by Bartal [6] or Fakcharoenphol et al. [14]) and choose an appropriate net for each cluster to be its portals, then the child portals of a cluster need not be a packing, because portals near the boundary of different clusters might be too close together. We resolve this by using Bartal's decomposition [6] a second time. After obtaining a standard decomposition, we apply the decomposition technique again to make minor adjustment to the boundaries of clusters. Here

is the main result that describes the properties of the hierarchical decomposition and portaling scheme.

THEOREM 4.1. (MAIN PARTITION-&-PORTAL THEOREM)
Given a metric (V, d) with $\dim_C = k$, and a parameter $\beta \leq 1$, there is a polynomial-time procedure that returns a probabilistic hierarchical partition of the metric with

(A1) The diameter of a height-i cluster is guaranteed to be at most $D_i + \beta D_{i-1}$, where $D_i = 4^i$.

(A2) The probability of (u, v) being separated at height i is at most $O(\log^2 n) \times \frac{d(u,v)}{D_i}$.

Moreover, each cluster C is equipped with a set of portals $U(C)$ such that the following properties hold:

(B1) For each non-root cluster C at height-i, the set of portals $U(C)$ forms a βD_i-covering of C.

(B2) Moreover, the set of portals in C and all its children form a $(\beta/4) D_{i-1}$-packing.

4.2.1 The Randomized Partitioning and Portaling Algorithm

Consider the metric (V, d) with unit minimum distance, and hence the aspect ratio being the diameter Δ of the metric. (Moreover, $\Delta \leq n/\varepsilon$, as noted before.) Let $H := 4$, and $L := \lceil \log_H(n/\varepsilon) \rceil$. Set $D_L := \Delta$, and $D_{i-1} := D_i/4$, as discussed before. We now give a hierarchical decomposition of (V, d) such that for each height-i cluster cluster C, the set $U(C)$ of portals is a is a βD_i-covering of C and its child portals is a $\frac{1}{4}\beta D_{i-1}$-packing, as described in the statement of Theorem 4.1.

Partitioning and Portaling Algorithm

---

1. Let $\mathcal{P}_L = \{V\}$ and $U(V) = \emptyset$.
2. For $i = L - 1$ down to 0,
   For each height-$(i + 1)$ cluster $C \in \mathcal{P}_{i+1}$,

   (a) Apply Bartal's probabilistic decomposition [6] on cluster C, using n as an upper bound on the number of points in C, such that the diameter of each resulting sub-cluster is at most D_i. This induces an initial partition $\widetilde{\mathcal{P}}_i$ on C.

   (b) Adjust the boundaries as follows:

   i. Note that $U(C)$ is a $\frac{1}{4}\beta D_{i+1}$-packing and $D_{i+1} = 4D_i$. Augment $U(C)$ to a βD_i-net $\widehat{U}(C)$ of C. Let Z be the set of points z in C that has no point in $\widehat{U}(C) \cap \widetilde{\mathcal{P}}_i(z)$ within distance βD_i.

   ii. Let $W := Z$, $X := C$, and $\overline{U}(C) := \emptyset$.

   iii. While W is non-empty,

   A) Pick any point u from W. Let $r := \beta D_i/4 \ln n$. Pick $z \in [0, \frac{1}{4}\beta D_i]$ randomly from the distribution $p(z) := \frac{n}{n-1} \cdot \frac{1}{r} e^{-z/r}$. Let $B := B(u, \frac{1}{4}\beta D_i + z)$.

B) If B contains some point c in $\widehat{U}(C)$, then all points in $B \cap X$ are moved to the height-i cluster currently containing c, otherwise, add u to $\overline{U}(C)$, and move all points in $B \cap X$ to the height-i cluster currently containing u.

C. Remove points in B from both X and W.

iv. Let the new partition on C be $\mathcal{P}_i$. For each new height-i cluster C', let $U(C') := C' \cap (\widehat{U}(C) \cup \overline{U}(C))$.

4.2.2 Properties of the Decomposition

The proofs of the following results are given in the full version.

LEMMA 4.1. (CORRECTNESS) *For $i < L$, for any height-$(i + 1)$ cluster C produced by the Decomposition Algorithm, then (1) for any child cluster C' of C, the set $U(C')$ is a βD_i-covering of C', and (2) the union of $U(C')$'s, over all the child clusters C' of C, is a $\frac{1}{4}\beta D_i$-packing.*

LEMMA 4.2. (SEPARATION PROBABILITY) *For each level i, $\Pr[(u, v)$ separated by $\mathcal{P}_i] \leq O(\log^2 n) \frac{d(u,v)}{D_i}$.*

In the following lemma, we use the definition of correlation dimension to bound the number of child portals in a cluster.

LEMMA 4.3. (SMALL NUMBER OF CHILD PORTALS) *Suppose the metric space (V, d) has correlation dimension at most k. For all clusters C, the union of $U(C')$ over all child clusters C' of C has size at most $(16/\beta + 4)^{k/2}\sqrt{n}$.*

Proof. Suppose cluster C is at height $i + 1$. By Lemma 4.1, the union S of $U(C')$ over all child clusters C' of C is a $\frac{1}{4}\beta D_i$-packing. Hence, it can be extended to a $\frac{1}{4}\beta D_i$-net N for the whole space V. Observe that from the construction, all points in C is contained in a ball with radius at most $(D_{i+1} + \beta D_i)/2$, though not necessarily centered at a point in N. Since N is a $4^i\beta$-net, C is contained in a ball of radius at most $D_{i+1} + \beta D_i$ centered at some net point $u \in N$. Hence, $S \subseteq \mathbf{B}_N(u, D_{i+1} + \beta D_i)$, which by Lemma 2.1 has size at most $(16/\beta + 4)^{k/2}\sqrt{|N|} \leq (16/\beta + 4)^{k/2}\sqrt{n}$.

4.3 The First TSP Algorithm

Using the partitioning and portaling scheme described in Section 4.2 and the dynamic program as described in [4], we have an algorithm for approximating TSP.

THEOREM 4.2. (THE FIRST TSP ALGORITHM)
There is a randomized algorithm for metric TSP, which for metrics with $\dim_C = k$, returns a tour of expected length at most $(1+\varepsilon)\mathsf{OPT}$ in time $2^{((\log n)/\varepsilon)^{O(k)}\sqrt{n}}$.

Proof. Since the aspect ratio of the metric is at most n/ε, the height of the decomposition is $L = O(\log \frac{n}{\varepsilon})$. By Theorem 4.1, each edge (u,v) of the optimal tour is cut at height-i with probability $\alpha \frac{d(u,v)}{D_i}$ with $\alpha = O(\log^2 n)$, in which case it suffers a length increase of $O(\beta D_i)$ to make it portal-respecting.

We set $\beta := O(\frac{\varepsilon}{L\alpha})$ to ensure that the total increase in length is at most $\alpha \beta L = \varepsilon d(u,v)$. Summing over all edges of OPT implies that the the expected length of this tour (and hence the length of the optimal portal-respecting tour) is at most $(1+\varepsilon)\mathsf{OPT}$.

We need to also bound the running time of the dynamic program: recall that an upper bound B for the number of portals in each cluster and its children would imply a $B^{O(B)}$ runtime.

By Lemma 4.3, it follows that $B \le (16/\beta+4)^{k/2}\sqrt{n}$. Hence, the runtime of the algorithm is $nL \cdot 2^{O(B \log B)} = \exp\{(\varepsilon^{-1}\log n)^{O(k)}\sqrt{n}\}$, as required. $\square$

4.4 Embedding into Small Treewidth Graphs

Observe that our probabilistic hierarchical decomposition procedure actually gives an embedding into a distribution of low treewidth graphs. Suppose we are given a particular hierarchical decomposition together with the portals for each cluster. We start with the complete weighted graph consistent with the metric, and delete any edge that is going out of a cluster but not via a portal. If the number of child portals for each cluster is at most B, then the treewidth of the resulting graph is at most B. From the padding property of the decomposition, the expected distortion of the distance between any pair of points is small. Using the same parameters as in the proof of Theorem 4.2, we have the following theorem.

THEOREM 4.3. (EMBEDDING INTO SMALL TREEWIDTH GRAPHS) *Given any constant $0 < \varepsilon < 1$ and k, metrics with correlation dimension at most k can be embedded into a distribution of graphs with treewidth $((\log n)/\varepsilon)^{O(k)}\sqrt{n}$ and distortion $1+\varepsilon$.*

5 A Sub-Exponential Time $(1 + \varepsilon)$-Approximation for TSP

In the previous section, we saw how to get a $(1 + \varepsilon)$-approximation algorithm for TSP on metrics with bounded correlation dimension, essentially using the idea of random embeddings into small treewidth graphs. The approach gives approximations for any problem on metric spaces which can be solved for small-treewidth graphs: however, it is limited by the fact that the $\sqrt{n}$-lollipop graph metric has bounded correlation dimension, and randomly $(1 + \varepsilon)$-approximating this graph requires the use of graphs with large treewidth.

In this section, we get an improved approximation for TSP using another useful observation. Consider the bad examples in Figure 1: the contribution to OPT due to the dense structure is much smaller than that from the low-dimensional ambient structure. For example, for the sub-grid with a $(1,2)$-TSP instance tacked onto it (Figure 1(b)), we can obtain a $(1 + \varepsilon)$-approximation to TSP on the grid (which contributes about $\Theta(n)$ to OPT), and stitch it together with a naïve 2-approximation to the hard instance (which only contributes $\Theta(\sqrt{n})$ to OPT). Of course, this is a simple case where the clustering is obvious; our algorithm must do some kind of clustering for all instances. Moreover, this indicates that we need to do a global accounting of cost: the sloppy approximation of the "hard" subproblem needs to be charged to the entire OPT, and not just the optimal tour on the subproblem.

Here are some of the issues we need to address (most of which are tied to each other), along with descriptions of how we handle them:

(a) **Avoiding Large Tables.** The immediate hurdle to a better runtime is that some cluster may have $\Theta(\sqrt{n})$ child portals and we have to spend $\sqrt{n}^{\sqrt{n}}$ time to compute the tables. Our idea here is to set a threshold B_0 such that in the dynamic program, if a cluster has more than $B > B_0$ portals among its children, we compute, in linear time, a tour on C that *only enters and leaves C once*, but now we incur an extra length of $O(B \times \mathrm{diam}(C))$ in the final tour we compute. In the sequel, we call this extra length the "MST-loss". This step implies that we need only spend $\min\{O(B), 2^{O(B_0 \log B_0)}\}$ time on any table computation. The patching procedure used here is reminiscent of the patching from [4], and is described in Section 5.2.

(b) **Paying for this Loss.** In contrast to previous works, the "MST-loss" due to patching cannot be charged locally, and hence we need to charge this to the cost of the global OPT. Using bounded correlation dimension, we can charge all the MST-losses over the entire run of the algorithm to $\varepsilon \mathsf{OPT}$.

(c) **A Potential Charging Scheme.** To be able to charge MST-losses in a global manner, we look at the hierarchical decomposition. The extra length incurred for patching height-i clusters is proportional to the number of child portals of the clusters to which patching is applied. *If the union of all the height-$(i-1)$ portals*

in the decomposition satisfied some packing condition, we could use Lemma 2.1 to bound the number of them, and hence the total MST-loss at height-i of the decomposition tree. However, the techniques developed so far (in Section 4.1) can only ensure that the child portals of a *single cluster* form a packing: we clearly need new techniques.

(d) A New Partitioning & Portaling Procedure. The method in the last section took a cluster C at height-$(i+1)$, cut it up, and then adjusted the boundaries of the subclusters created at height-i to ensure that the union of the portals in these subclusters formed a packing. However, the portals in all the grand-children of C (i.e., all the clusters at height-$(i-1)$ below C) may not form a packing: hence we have to re-adjust the boundaries created at height-i yet again. In fact, when clusters at a certain level are created, the boundaries for clusters in all higher levels have to be readjusted. This can potentially increase the probability that a pair of points are separated at each level. This is resolved by ensuring that cluster diameters fall by logarithmic factors instead of by constants.

(e) Avoiding Computation of Correlation Dimension. As given in Theorem 2.2, it is hard to approximate the correlation dimension of a given metric. However, the algorithm can guess the correlation dimension k of the input metric. It starts from small values of k and for each net encountered, it takes polynomial time to verify the bounded average growth rate property (2.2). Whenever property (2.2) is violated for some net, we know the current estimation of the correlation dimension is too small. The value of k is increased and the algorithm is restarted. Since the correlation dimension is at most $O(\log n)$ and the running time is doubly exponential in k, the extra time incurred for trying out smaller values of k would not affect the asymptotic running time.

The general framework of using hierarchical decomposition and portals to approximate the TSP still applies here. We give the properties of a more sophisticated partitioning and portaling scheme in Section 5.1, and analyze the MST-loss incurred from patching in Section 5.2.

5.1 The Modified Partitioning and Portaling Algorithm

The main difference is that when a height-i partition is performed, all higher height partitions are modified, in order to ensure that all height-i portals form a packing. Let $H \geq 4$ be a parameter (possibly depending on n) that will be determined later. Let $L := \lceil \log_H(n/\varepsilon) \rceil$. Set $D_L := \Delta$, the diameter of (V, d); $D_{i-1} := D_i/H$. As before, the diameter of a height-

i cluster is at most D_i. The details of the modified construction appear in the full version.

Properties of the Modified Decomposition Scheme. The proofs of the following lemmas appear in the full version.

LEMMA 5.1. *The probability that a pair (u, v) of points is separated by the height-i partition $\mathcal{P}_i$ is at most* $(4t \log n)^L \cdot \frac{d(u,v)}{D_i} = O(\log n)^L \cdot \frac{d(u,v)}{D_i}$.

LEMMA 5.2. *For each height i, the set U_i of height-i portals is a $\frac{1}{4}\beta D_{i-1}$-packing and for each height-i cluster C, the set $U_i \cap C$ of portals is a βD_i-covering of C.*

5.2 Handling Large Portal Sets via Patching

(A) Patching a single cluster. If a cluster C has many child portals (say about $\sqrt{n}$ portals), it is too expensive to compute the entries corresponding to C. In particular, computing the standard TSP table for this cluster would require $O(\sqrt{n}^{\sqrt{n}}) = 2^{\tilde{O}(\sqrt{n})}$ time, which in itself would dash all hopes of obtaining a sub-exponential time algorithm. To avoid this, we do a two step patching described in the following.

The first idea is simple: we can convert any given tour such that it enters the cluster C through some portal x, performs a traveling salesperson tour on points in cluster C, and leaves cluster C through x, incurring an extra length of BD. However, *computing such a tour within the cluster C requires work as well, and we need to ensure that this computation can be done fast:* if cluster C has too many child portals, it would be too expensive to compute the optimal tour inside C. Hence, we need a second patching step. We defer the details to the full version and state the result of the patching process.

PROPOSITION 5.1. *Suppose cluster C has diameter D, and that there are at most B child portals in the cluster C. Then, given any tour on the vertices V, the tour can be modified such that it enters and leaves the cluster C through a* single portal*; moreover, such a tour within cluster C can be computed in time $O(B)$, from the entries in the child clusters of C, such that the total cost of patching cluster C is at most $2BD$.*

(B) Patching All Dense Clusters. We set a threshold B_0 and apply patching to all clusters having more than B_0 child portals. Using the fact that the correlation dimension is bounded and from Lemma 5.2 that all the portals in one level form a packing, we can bound the cost of patching. We defer the details to the full version and state the following result.

LEMMA 5.3. *Suppose the metric has correlation dimension at most k and patching is applied whenever a cluster has more than B_0 child portals. Then, the total extra cost incurred by patching is at most $\frac{L}{B_0}(\frac{8H}{\beta})^{k+1}\mathsf{OPT}$.*

5.3 The Second TSP Algorithm

THEOREM 5.1. (SUB-EXPONENTIAL TIME ALGORITHM FOR TSP) *For any metric with correlation dimension k, we can give a randomized $(1+\varepsilon)$-approximation for TSP in time $\exp\{(\varepsilon^{-1}2^{\sqrt{\log n \log\log n}})^{4k}\} = 2^{O_{\varepsilon,k}(n^\delta)}$, for any $\delta > 0$.*

Proof. We create a probabilistic hierarchical decomposition, where the diameter at height-i is $D_i = H^i$ for some parameter $H \geq 4$. Hence the depth of the tree is $L := \Theta(\log_H(n/\varepsilon))$. As indicated above (and proved in Lemma 5.1), the probability that (u,v) are separated at level-i is at most $\alpha\frac{d(u,v)}{D_i}$, with $\alpha = O(\log n)^L$. Moreover, portals in clusters of diameter D_i form a βD_i-covering and since there are L levels, the total increase in the TSP length is $O(\alpha\beta L)\mathsf{OPT}$. To make this at most $\varepsilon/2$, we set $\beta = O(\varepsilon/L\alpha)$.

Finally, from Lemma 5.3, the length increase from patching (the "MST-loss") is $\frac{L}{B_0}(\frac{8H}{\beta})^{k+1}\mathsf{OPT}$. To make this at most $\varepsilon/2$ as well, we set pick B_0 such that $\frac{L}{B_0}(\frac{8H}{\beta})^{k+1} = \varepsilon/2$.

The only parameter left to be chosen is H. Observe that the running time depends on B_0 and so H is chosen to minimize B_0. Note that $B_0 = (\frac{L}{\varepsilon})^{k+2}O(H\alpha)^{k+1}$. Observe that $H\alpha$ is the dominating term, and also that as H increases, α decreases. It happens that in this case the best value is attained when $H = \alpha$. This is satisfied when $\log H = \sqrt{\log\frac{n}{\varepsilon}\log\log n}$.

It follows that it suffices to set the threshold $B_0 = \varepsilon^{-(k+1)}2^{2(k+1)\sqrt{\log\frac{n}{\varepsilon}\log\log n}} = (\varepsilon^{-1}\cdot 2^{\sqrt{\log n \log\log n}})^{3k}$, recalling $\varepsilon > \frac{1}{n}$. Hence, we obtain a tour with expected length $(1+\varepsilon)$ times that of the optimal tour in time $nL \cdot 2^{O(B\log B)}$, which for the above settings of parameters is

$$\exp\{(\varepsilon^{-1}\cdot 2^{\sqrt{\log n \log\log n}})^{4k}\} = 2^{O_{\varepsilon,k}(n^\delta)},$$

for any $\delta > 0$.

6 Summary and Conclusions

We have considered a global notion of dimension, which tries to capture the "average" complexity of metrics: our notion of correlation dimension captures metrics that potentially contain small dense clusters (of size up to $O(\sqrt{n})$) but have small average growth-rate. We show that metrics with a low correlation dimension do indeed admit efficient algorithms for a variety of problems.

Many questions remain open: can we improve the running time of our algorithm for TSP? A more open-ended question is defining other notions of dimension for metric spaces: it is fairly unlikely that one notion can capture the complexity of metrics (both the local complexity, as in doubling, as well as the global behavior). Since one definition may not fit all situations, it seems reasonable to consider several definitions, whose properties can then be exploited under the appropriate circumstances.

References

[1] Ittai Abraham, Cyril Gavoille, Andrew Goldberg, and Dahlia Malkhi. Routing in networks with low doubling dimension. In *26th ICDCS*, page 75, 2006.

[2] Ittai Abraham and Dahlia Malkhi. Name independent routing for growth bounded networks. In *17th SPAA*, pages 49–55, 2005.

[3] Stefan Arnborg and Andrzej Proskurowski. Linear time algorithms for NP-hard problems restricted to partial k-trees. *Discrete Appl. Math.*, 23(1):11–24, 1989.

[4] S. Arora. Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems. *J. ACM*, 45(5):753–782, 1998.

[5] Sanjeev Arora, Prabhakar Raghavan, and Satish Rao. Approximation schemes for Euclidean k-medians and related problems. In *STOC '98 (Dallas, TX)*, pages 106–113. ACM, New York, 1999.

[6] Yair Bartal. Probabilistic approximations of metric spaces and its algorithmic applications. In *37th FOCS*, pages 184–193, 1996.

[7] Alberto Belussi and Christos Faloutsos. Estimating the selectivity of spatial queries using the 'correlation' fractal dimension. In *21st VLDB*, pages 299–310, 1995.

[8] Alina Beygelzimer, Sham Kakade, and John Langford. Cover trees for nearest neighbor. In *23rd International Conference on Machine Learning*, pages 97–104, 2006.

[9] Hubert T-H. Chan, Anupam Gupta, Bruce M. Maggs, and Shuheng Zhou. On hierarchical routing in DOubling metrics. In *16th SODA*, pages 762–771, 2005.

[10] Kenneth L. Clarkson. Nearest-neighbor searching and metric space dimensions. In G. Shakhnarovich, T. Darrell, and P. Indyk, editors, *Nearest-Neighbor Methods for Learning and Vision: Theory and Practice*, pages 15–59. MIT Press, 2006.

[11] Richard Cole and Lee-Ad Gottlieb. Searching dynamic point sets in spaces with bounded doubling dimension. In *38th STOC*, pages 574–583, 2006.

[12] Artur Czumaj and Andrzej Lingas. Fast approximation schemes for Euclidean multi-connectivity problems (extended abstract). In *Automata, languages and programming (Geneva, 2000)*, volume 1853 of *Lecture Notes in Comput. Sci.*, pages 856–868. Springer, Berlin, 2000.

[13] Artur Czumaj, Andrzej Lingas, and Hairong Zhao. Polynomial-time approximation schemes for the Euclidean survivable network design problem. In *Automata, languages and programming*, volume 2380 of *Lecture Notes in Comput. Sci.*, pages 973–984. Springer, Berlin, 2002.

[14] Jittat Fakcharoenphol, Satish Rao, and Kunal Talwar. A tight bound on approximating arbitrary metrics by tree metrics. *J. Comput. System Sci.*, 69(3):485–497, 2004.

[15] Christos Faloutsos and Ibrahim Kamel. Beyond uniformity and independence: Analysis of r-trees using the concept of fractal dimension. In *13th PODS*, pages 4–13, 1994.

[16] Peter Grassberger and Itamar Procaccia. Measuring the strangeness of strange attractors. *Physica D*, 9:189–208, 1983.

[17] Michelangelo Grigni. Approximate tsp in graphs with forbidden minors. In *27st ICALP*, pages 869–877, 2000.

[18] Anupam Gupta, Robert Krauthgamer, and James R. Lee. Bounded geometries, fractals, and low–distortion embeddings. In *44th FOCS*, pages 534–543, 2003.

[19] Sariel Har-Peled and Manor Mendel. Fast constructions of nets in low dimensional metrics, and their applications. In *21st SOCG*, pages 150–158, 2005.

[20] Johan Håstad. Clique is hard to approximate within $n^{1-\epsilon}$. In *37th FOCS*, pages 627–636, 1996.

[21] K. Hildrum, J. D. Kubiatowicz, S. Rao, and B. Y. Zhao. Distributed object location in a dynamic network. In *14th SPAA*, pages 41–52, August 2002.

[22] David R. Karger and Matthias Ruhl. Finding nearest neighbors in growth-restricted metrics. In *34th STOC*, pages 63–66, 2002.

[23] Philip N. Klein. A linear-time approximation scheme for planar weighted tsp. In *46th FOCS*, pages 647–657, 2005.

[24] Jon M. Kleinberg, Aleksandrs Slivkins, and Tom Wexler. Triangulation and embedding using small sets of beacons. In *45th FOCS*, pages 444–453, 2004.

[25] Leonard Kleinrock and Farouk Kamoun. Hierarchical routing for large networks. Performance evaluation and optimization. *Comput. Networks*, 1(3):155–174, 1976/77.

[26] Stavros G. Kolliopoulos and Satish Rao. A nearly linear-time approximation scheme for the Euclidean k-median problem. In *Algorithms—ESA '99 (Prague)*, volume 1643 of *Lecture Notes in Comput. Sci.*, pages 378–389. Springer, Berlin, 1999.

[27] Goran Konjevod, Andréa W. Richa, and Donglin Xia. On sampling in higher-dimensional peer-to-peer systems. In *LATIN 2006: Theoretical informatics*, volume 3887 of *Lecture Notes in Comput. Sci.*, pages 641–652, Berlin, 2006. Springer.

[28] Goran Konjevod, Andréa W. Richa, and Donglin Xia. Optimal-stretch name-independent compact routing in doubling metrics. In *25th PODC*, pages 198–207, 2006.

[29] Robert Krauthgamer and James R. Lee. Navigating nets: simple algorithms for proximity search. In *15th SODA*, pages 798–807, 2004.

[30] Robert Krauthgamer and James R. Lee. The black-box complexity of nearest-neighbor search. *Theoret. Comput. Sci.*, 348(2-3):262–276, 2005.

[31] F. Kuhn, T. Moscibroda, and R. Wattenhofer. On the locality of bounded growth. In *24th PODC*, pages 60–68, 2005.

[32] Joseph S. B. Mitchell. Guillotine subdivisions approximate polygonal subdivisions: a simple polynomial-time approximation scheme for geometric TSP, k-MST, and related problems. *SIAM J. Comput.*, 28(4):1298–1309, 1999.

[33] Bernd-Uwe Pagel, Flip Korn, and Christos Faloutsos. Deflating the dimensionality curse using multiple fractal dimensions. In *ICDE*, pages 589–598, 2000.

[34] Christos H. Papadimitriou and Mihalis Yannakakis. The traveling salesman problem with distances one and two. *Math. Oper. Res.*, 18(1):1–11, 1993.

[35] C. G. Plaxton, R. Rajaraman, and A. W. Richa. Accessing nearby copies of replicated objects in a distributed environment. *Theory Comput. Syst.*, 32(3):241–280, 1999. ACM Symposium on Parallel Algorithms and Architectures (Padua, 1996).

[36] Satish B. Rao and Warren D. Smith. Approximating geometrical graphs via "spanners" and "banyans". In *STOC '98 (Dallas, TX)*, pages 540–550. ACM, New York, 1999.

[37] Aleksandrs Slivkins. Distance estimation and object location via rings of neighbors. In *24th PODC*, pages 41–50, 2005.

[38] Kunal Talwar. Bypassing the embedding: Algorithms for low-dimensional metrics. In *36th STOC*, pages 281–290, 2004.

[39] Luca Trevisan. When Hamming meets Euclid: the approximability of geometric TSP and Steiner tree. *SIAM J. Comput.*, 30(2):475–485, 2000.