2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)

On Light Spanners, Low-treewidth Embeddings and Efficient Traversing in Minor-free Graphs

Vincent Cohen-Addad
Google Research
vcohenad@gmail.com

Arnold Filtser
Columbia University
Computer Science Department
arnold273@gmail.com

Philip N. Klein
Brown University
Department of Computer Science
klein@brown.edu

Hung Le
University of Massachusetts at Amherst
College of Information & Computer Sciences
hungle@cs.umass.edu

***Abstract*—Understanding the structure of minor-free metrics, namely shortest path metrics obtained over a weighted graph excluding a fixed minor, has been an important research direction since the fundamental work of Robertson and Seymour. A fundamental idea that helps both to understand the structural properties of these metrics and lead to strong algorithmic results is to construct a "small-complexity" graph that approximately preserves distances between pairs of points of the metric. We show the two following structural results for minor-free metrics:**

1) **Construction of a *light* subset spanner. Given a subset of vertices called terminals, and ϵ, in polynomial time we construct a subgraph that preserves all pairwise distances between terminals up to a multiplicative $1+\epsilon$ factor, of total weight at most $O_\epsilon(1)$ times the weight of the minimal Steiner tree spanning the terminals.**
2) **Construction of a stochastic metric embedding into low treewidth graphs with expected additive distortion ϵD. Namely, given a minor-free graph $G=(V,E,w)$ of diameter D, and parameter ϵ, we construct a distribution $\mathcal{D}$ over dominating metric embeddings into treewidth-$O_\epsilon(\log n)$ graphs such that $\forall u,v\in V$, $\mathbb{E}_{f\sim\mathcal{D}}[d_H(f(u),f(v))]\le d_G(u,v)+\epsilon D$.**

Our results have the following algorithmic consequences: (1) the first efficient approximation scheme for subset TSP in minor-free metrics; (2) the first approximation scheme for bounded-capacity vehicle routing in minor-free metrics; (3) the first efficient approximation scheme for bounded-capacity vehicle routing on bounded genus metrics. En route to the latter result, we design the first FPT approximation scheme for bounded-capacity vehicle routing on bounded-treewidth graphs (parameterized by the treewidth).

***Keywords*-travelling salesperson problem; minor-free graphs; vehicle routing; metric embedding; spanners;**

I. Introduction

Fundamental routing problems such as the Traveling Salesman Problem (TSP) and the Vehicle Routing Problem have been widely studied since the 50s. Given a metric space, the goal is to find a minimum-weight collection of tours (only one for TSP) so as to meet a prescribed demand at some points of the metric space. The research on these problems, from both practical and theoretical perspectives, has been part of the agenda of the operations research and algorithm-design communities for many decades.Both problems have been the source of inspiration for many algorithmic breakthroughs and remain good examples of the limits of the power of our algorithmic methods.

Since both problems are APX-hard in general graphs [1, 2], it has been a natural and successful research direction to focus on *structured* metric spaces. Initially, researchers focused on achieving polynomial-time approximation schemes (PTASs) for TSP in planar graphs [3, 4] and Euclidean metrics [5, 6]. Two themes emerged in the ensuing research: speed-ups and generalization.

In the area of speed-ups, a long line of research on Euclidean TSP improved the running time $n^{O(1/\epsilon)}$ of the initial algorithm by Arora to linear time [7]. In a parallel research thread, Klein [8, 9] gave the first *efficient PTAS*[1] for TSP in weighted planar graphs, a linear-time algorithm.

In the area of generalization, a key question was whether these results applied to more general (and more abstract) families of metrics. One such generalization of Euclidean metrics is metrics of bounded doubling dimension. Talwar [10] gave a *quasi*-polynomial-time approximation scheme (QPTAS) for this problem which was then improved to an EPTAS [11]. In minor-free metrics, an important generalization of planar metrics, Grigni [12] gave a QPTAS for TSP which was recently improved to EPTAS by Borradaile et al. [13].

[1] A PTAS is an *efficient* PTAS (an EPTAS) if its running time is bounded by a polynomial n^c whose degree c does not depend on ϵ

2575-8454/20/$31.00 ©2020 IEEE
DOI 10.1109/FOCS46700.2020.00061

When the metric is that of a planar/minor-free graph, the problem of visiting every vertex is not as natural as that of visiting a given subset of vertices (the *Steiner TSP* or *subset TSP*) since the latter cannot be reduced to the former without destroying the graph structure. The latter problem turns out to be much harder than TSP in minor-free graphs, and in fact no approximation scheme was known until the recent PTAS for subset TSP by Le [14]. This immediately raises the question:

Question 1. *Is there an EPTAS for subset TSP in minor-free graphs?*

The purpose of this line of work is to understand what are the most general metrics for which we can obtain approximation schemes for routing problems, and when it is the case how fast can the approximation schemes be made. Toward this goal, minor-free metrics have been a testbed of choice for generalizing the algorithmic techniques designed for planar or bounded-genus graphs. Indeed, while minor-free metrics offer very structured decompositions, as shown by the celebrated work of Robertson and Seymour [15], Klein et al. [16], and Abraham et al. [17] (see also [18, 19]), they do not exhibit a strong topological structure. Hence, various strong results for planar metrics, such as the efficient approximation schemes for Steiner Tree [20] or Subset TSP [21], are not known to exist in minor-free metrics.

Space	**Lightness**	**TSP runtime**	**Reference**
$(\mathbb{R}^{O(1)}, \|\cdot\|_2)$	$\epsilon^{-O(1)}$	$2^{\epsilon^{-O(1)}} \cdot \tilde{O}(n)$	[22, 23]
Doubling $O(1)$	$\epsilon^{-O(1)}$	$2^{\epsilon^{-O(1)}} \cdot \tilde{O}(n)$	[11, 24]
Planar	$O(1/\epsilon)$	$2^{O(1/\epsilon^2)} \cdot O(n)$	[25, 8]
$K_{O(1)}$ free	$\tilde{O}(1/\epsilon^3)$	$2^{\tilde{O}(1/\epsilon^4)} \cdot n^{O(1)}$	[26, 13]

A common ingredient to designing efficient PTAS for TSP is the notion of *light spanner*: a weighted subgraph H over the points of the original graph/metric space G that preserves all pairwise distances up to some $1+\epsilon$ multiplicative factor (i.e. $\forall u, v \in V(G)$, $d_H(u,v) \leq (1+\epsilon) \cdot d_G(u,v)$). The *lightness* of the spanner H is the ratio between the total weight of H and that of the Minimum Spanning Tree (MST) of G. While significant progress has been made on understanding the structure of spanners (see the table), it is not the case for *subset spanners*. A subset spanner H w.r.t. a prescribed subset K of vertices, called *terminals*, is a subgraph that preserves distances between terminals up to a $1+\epsilon$ multiplicative factor (i.e. $\forall u, v \in K$, $d_H(u,v) \leq (1+\epsilon) \cdot d_G(u,v)$). The *lightness* of H is the ratio between the weight of H and the weight of a minimum Steiner tree[2] w.r.t. K. While for light spanners the simple greedy algorithm is "existentially optimal" [27], in almost all settings, no such "universal" algorithm is known for constructing light subset spanners. In planar graphs, Klein [21] constructed the first light subset spanner. Borradaile et al. [28] generalized Klein's construction to bounded-genus graphs. Unfortunately, generalizing these two results to minor-free metrics remained a major challenge since both approaches relied on topological arguments. Recently, Le [14] gave the first polynomial-time algorithm for computing a subset spanner with lightness $\text{poly}(\frac{1}{\epsilon}) \cdot \log |K|$ in K_r-minor-free graphs. However, the following question remains a fundamental open problem, often mentioned in the literature [26, 28, 13, 14].

Question 2. *Does a subset spanner of lightness* $\text{poly}(\frac{1}{\epsilon})$ *exist in minor-free graphs?*

A related routing problem is the vehicle routing problem. Given a *capacity* Q and a graph with weights and a special vertex called the *depot*, and given an assignment of nonnegative *demands* to vertices, the goal is to find a minimum-weight collection of tours that start and end at the depot such that each vertex with nonzero demand is assigned to a tour that visits it, and the total demand assigned to each tour is at most Q. This is a classic routing problem, introduced in the late 50s by Dantzig and Ramser [29]. While major progress has been made on TSP during the 90s and 00s for planar and Euclidean metrics, the current understanding of vehicle routing is much less satisfactory. If the capacity and the demands are arbitrary nonnegative integers, the problem is APX-hard for trees by reduction from the partition problem. For unit demands in the Euclidean plane, Das and Mathieu [30] gave a quasi-PTAS. For arbitrary graphs, the problem remains NP-hard when the capacity Q is bounded by a constant and the demands are unit [2]. In view of the popularity of product-delivery services, the bounded-capacity problem is still interesting. For bounded capacity and unit demands, Asano et al. gave an efficient approximation scheme [2] for the Euclidean plane. For bounded capacity in planar graphs (but where capacities are not necessarily unit), Becker et al. [32] gave a quasi-polynomial approximation scheme, which was subsequently improved to a running time of $n^{(Q\epsilon)^{-O(Q/\epsilon)}}$ [33]. This raises the following question:

Question 3. *Does bounded-capacity vehicle routing admit an EPTAS in planar and bounded-genus graphs?*

Since the techniques in previous work [32, 33] for the bounded-capacity vehicle routing problem rely on topological arguments, they are not extensible to minor-free graphs. In fact, no nontrivial approximation scheme was known for this problem in minor-free graphs. We ask:

[2] A Steiner tree is a connected subgraph containing all the terminals K. A minimum Steiner tree is a minimum-weight such subgraph; because cycles do no help in achieving connectivity, we can require that the subgraph be a tree.

Question 4. *Is it possible to design a quasi-polynomial-time approximation scheme for bounded-capacity vehicle routing in minor-free graphs?*

The approach of Becker et al. (drawing on [34]) is through *metric embeddings*, similar to the celebrated work of Bartal [35] and Fakcharoenphol et al. [36] who showed how to embed any metric space into a simple tree-like structure. Specifically, Becker et al. aim at embedding the input metric space into a "simpler" target space, namely a graph of bounded treewidth, while (approximately) preserving all pairwise distances. A major constraint arising in this setting is that for obtaining approximation schemes, the distortion of the distance should be carefully controlled. An ideal scenario would be to embed n-vertex minor free graphs into graphs of treewidth at most $O_\epsilon(\log n)$, while preserving the pairwise distance up to a $1+\epsilon$ factor. Unfortunately, as implied by the work of Chakrabarti et al. [37], there are n vertex planar graphs such that every (stochastic) embedding into $o(\sqrt{n})$-treewidth graphs must incur expected multiplicative distortion $\Omega(\log n)$ (see also [38, 39, 40] for embeddings into Euclidean metrics).

Bypassing the above roadblock, Eisenstat et al. [34] and Fox-Epstein et al. [41] showed how to embed planar metrics into bounded-treewidth graphs while preserving distances up to a controlled *additive* distortion. Specifically, given a planar graph G and a parameter ϵ, they showed how to construct a metric embedding into a graph H of bounded treewidth such that all pairwise distances between pairs of vertices are preserved up to an additive ϵD factor, where D is the diameter of G. While ϵD may look like a crude additive bound, it is good enough for obtaining approximation schemes for some classic problems such as k-center and vehicle routing. While Eisenstat et al. constructed an embedding into a graph of treewidth $\mathrm{poly}(\frac{1}{\epsilon})\cdot\log n$, Fox-Epstein et al. constructed an embedding into a graph of treewidth $\mathrm{poly}(\frac{1}{\epsilon})$, leading to the first PTAS for vehicle routing (with running time $n^{(Q/\epsilon)^{O(Q/\epsilon)}}$). Yet for minor-free graphs, or even bounded-genus graphs, obtaining such a result with any non-trivial bound on the treewidth is a major challenge; the embedding of Fox-Epstein et al. [41] heavily relies on planarity (for example by using the face-vertex incident graph). Therefore, prior to our work, the following question is open.

Question 5. *Is it possible to (perhaps stochastically) embed a minor-free graph with diameter D to a graph with treewidth $\mathrm{polylog}(n)$ and additive distortion at most ϵD?*

A. Main contribution

We answer all the above questions by the affirmative. Our first main contribution is a "truly" *light subset spanner* for minor-free metrics that bridges the gap for spanners between planar and minor-free metrics; this completely settles 2. In the following, the O_r notation hides factors in r, e.g. $x = O_r(m) \iff x \le m \cdot f(r)$ for some sufficiently large m and computable function f; and $\mathrm{poly}(x)$ is (some) polynomial function of x.

Theorem 1. *There is a polynomial time algorithm that, given a K_r-minor-free graph G, a set of terminals $K \subseteq V(G)$, and a parameter $\epsilon \in (0,1)$, computes a subset spanner with distortion $1+\epsilon$ and lightness $O_r(\mathrm{poly}(\frac{1}{\epsilon}))$.*

Our second main contribution is a *stochastic embedding* of minor-free graphs into bounded-treewidth graphs with small expected additive distortion, obtaining the first result of this kind for minor-free graphs and resolving Question 5 positively.

Theorem 2. *There is a randomized polynomial-time algorithm that, given an n-vertex K_r-minor-free graph G of diameter D, and a parameter $\epsilon \in (0,1)$, constructs a stochastic embedding from G into graphs with treewidth $O_r(\frac{\log n}{\epsilon^2})$, and expected additive distortion ϵD.*

While the embedding of planar graphs to low treewidth graphs by Fox-Epstein et al. [41] is deterministic, our embedding in Theorem 2 is stochastic. Thus, it is natural to ask whether randomness is necessary. We show in Theorem 3 below that the embedding must be stochastic to guarantee (expected) additive distortion ϵD, for small enough ϵ .

Theorem 3. *There is an infinite graph family $\mathcal{H}$ of K_6-free graphs, such that for every $H \in \mathcal{H}$ with n vertices and diameter D, every dominating embedding of H into a treewidth-$o(\sqrt{n})$ graph has additive distortion at least $\frac{1}{20}\cdot D$.*

For the more restricted case of a graph with genus g, we can construct a deterministic embedding without any dependence on the number of vertices.

Theorem 4. *Given a genus-g graph G of diameter D, and a parameter $\epsilon \in (0,1)$, there exists an embedding f from G to a graph H of treewidth $O_g(\mathrm{poly}(\frac{1}{\epsilon}))$ with additive distortion ϵD.*

Next we describe the algorithmic consequences of our results. First, we obtain an efficient PTAS for the Subset TSP problem in K_r-minor-free graphs for any fixed r, thereby answering Question 1.

Theorem 5. *There is an algorithm that, given a set of terminals K in an n-vertex K_r-minor-free graph G, runs in time $2^{O_r(\mathrm{poly}(1/\epsilon))}n^{O(1)}$ and finds a tour that* *visits every vertex in K and that has length at most $1+\epsilon$ times the length of the shortest tour.*

Second, we obtain the first polynomial-time approximation scheme for bounded-capacity vehicle routing in K_r-minor-free graphs.

Theorem 6. *There is a randomized algorithm that, given an n-vertex K_r-minor-free graph G and an instance of bounded-capacity vehicle routing on G, in time $n^{O_{\epsilon,Q,r}(\log\log n)}$ returns a solution with expected cost at most $1+\epsilon$ times the cost of the optimal solution.*

Theorem 6 provides an answer to Question 4. En route to this result, we design a new dynamic program for bounded-capacity vehicle routing on bounded-treewidth graphs that constitutes the first approximation scheme that is fixed-parameter tractable in the treewidth (and also in ϵ) for this class of graphs. For planar graphs and bounded-genus graphs, this yields a $2^{\text{poly}(1/\epsilon)}n^{O(1)}$ approximation scheme and answers Question 3.

Theorem 7. *There is a randomized algorithm that, given a graph G with genus at most g and an instance of bounded-capacity vehicle routing on G, in time $2^{O_{g,Q}(\text{poly}(1/\epsilon))}n^{O(1)}$ returns a solution whose expected cost at most $1+\epsilon$ times the cost of the optimal solution.*

A tool in our algorithm in Theorem 7 is a new efficient dynamic program for approximating bounded-capacity vehicle routing in bounded-treewidth graphs. The best exact algorithm known for bounded-treewidth graphs has running time $n^{O(Q\text{tw})}$ [42].

Theorem 8. *Let $\text{tw}, \epsilon > 0$. There is an algorithm that, for any instance of the vehicle routing problem (G, Q, s) such that G has treewidth tw and n vertices, outputs a $(1+\epsilon)$-approximate solution in time $(Q\epsilon^{-1}\log n)^{O(Q\text{tw}/\epsilon)}n^{O(1)}$.*

B. Techniques

In their seminal series of papers regarding minor free graph, Robertson and Seymour showed how to decompose a minor-free graph into four "basic components": *surface-embedded graphs*, *apices*, *vortices* and *clique-sums* [15]. Their decomposition suggested an algorithmic methodology, called the *RS framework*, for solving a combinatorial optimization problem on minor-free graphs: solve the problem on planar graphs, and then generalize to bounded-genus graphs, to graphs embedded on a surface with few vortices, then deal with the apices, and finally extend to minor-free graphs. The RS framework has been successfully applied to many problems such as vertex cover, independent set and dominating set [43, 44]. A common feature for these problems was that the graphs were unweighted, and the problems rather "local". This success can be traced back to the pioneering work of Grohe [43] who showed how to handle graphs embedded on a surface with few vortices by showing that these graphs have linear local treewidth.

However, there is no analogous tool that can be applied to fundamental connectivity problems such as Subset TSP, Steiner tree, and survivable network design. Therefore, even though efficient PTASes for these problems were known for planar graphs [21, 45, 46] for a long time, achieving similar results for any of them in minor-free graphs remained an open problem. Guided by the RS framework, we propose a multi-step framework for light subset spanner and embedding problems in minor-free graphs.

A multi-step framework: The fundamental building block in our framework is planar graphs each with a *single vortex* with bounded diameter D, on which we solve the problems (Step 1 in our framework). We consider this as a major conceptual contribution as we overcome the barrier posed by vortices. We do so by introducing a hierarchical decomposition where each cluster in every level of the decomposition is separated from the rest of the graph by a constant number of shortest paths *of the input graph*.[3] Similar decomposition for planar graphs [4, 48] and bounded-genus graphs [49] has found many algorithmic applications [4, 50, 34, 49]. Surprisingly, already for the rather restricted case of apex graphs,[4] it is impossible to have such a decomposition. We believe that our decomposition is of independent interest.

While it is clear that the diameter parameter D is relevant for the embedding problem, a priori it is unclear why it is useful for the light-subset-spanner problem. As we will see later, the diameter comes from a reduction to subset local spanners (Le [14]), while the assumption is enabled by using sparse covers [51].

In Step 2, we generalize the results to K_r-minor-free graphs. Step 2 is broken into several mini-steps. In Mini-Step 2.1,[5] we handle the case of planar graphs with more than one vortex; we introduce a *vortex-merging operation* to reduce to the special case in Step 1. In Mini-Step 2.2, we handle graphs embedded on a surface with multiple vortices. The idea is to cut along vortex paths to reduce the genus one at a time until the surface embedded part is planar (genus 0), and in this case, Step 2.1 is applicable. In Mini-Step 2.3,

[3] One might hope that a similar decomposition can be constructed using the shortest-path separator of Abraham and Gavoille [47] directly. Unfortunately, this is impossible as the length of the shortest paths in [47] is unbounded w.r.t. D. Rather, they are shortest paths in different subgraphs of the original graph.

[4] A graph G is an apex graph if there is a vertex v such that $G \setminus v$ is a planar graph.

[5] In the subset spanner problem, there is an additional step where we remove the constraint on the diameter of the graph, and this becomes Step 2.0.

we handle graphs embedded on a surface with multiple vortices and *a constant number of apices*, a.k.a *nearly embeddable graphs*. In Mini-Step 2.4, we show how to handle general K_r-minor-free graphs by dealing with clique-sums.

In this multi-step framework, there are some steps that are simple to implement for one problem but challenging for the other. For example, implementing Mini-Step 2.3 is simple in the light subset spanner problem, while it is highly non-trivial for the embedding problem; removing apices can result in a graph with unbounded diameter. Novel ideas are typically needed to resolve these challenges; we refer the reader to Section III for more technical details.

We believe that our multi-step framework will find applications in designing PTASes for other problems in K_r-minor-free graphs, such as minimum Steiner tree or survivable network design.

An FPT approximation scheme for vehicle routing on low-treewidth graphs : Our $(1+\epsilon)$-approximation for vehicle routing with bounded capacity in bounded treewidth graphs relies on a dynamic program that proceeds along the clusters of a branch decomposition,[6] namely the subgraphs induced by the leaves of the subtrees of the branch decomposition. One key idea is to show that there exists a near-optimal solution such that the number of tours entering (and leaving) a given cluster with some fixed capacity $q \in [Q]$ can be rounded to a power of $1+\tilde{\epsilon}$, for some $\tilde{\epsilon}$ to be chosen later. To achieve this, we start from the optimum solution and introduce *artificial paths*, namely paths that start at a vertex and go to the depot (or from the depot to a vertex), without making any delivery and whose only purpose is to help *rounding* the number of paths entering or leaving a given cluster of the decomposition (i.e.: making it a power of $1+\tilde{\epsilon}$). This immediately reduces the number of entries in the dynamic programming table we are using, reducing the running time of the dynamic program to the desired complexity.

The main challenge becomes to bound the total cost of artificial paths hence created so as to show that the obtained solution has cost at most $1+\epsilon$ times the cost of the optimum solution. To do so, we design a charging scheme and prove that every time a new path is created, its cost can be charged to the cost of some $\tilde{\epsilon}^{-1}$ paths of the original optimum solution. Then, we ensure that each path of the original optimum solution does not get charged more than ϵ times. This is done by showing by defining that a path *enters* (resp. *leaves*) a cluster only if it is making its next delivery (resp. it has made its last delivery) to a vertex inside. This definition helps limit the number of times a path gets charged to $\tilde{\epsilon} = \epsilon/(Q\log n)$ but it also separates the underlying shortest path metric from the structure of the graph: A path from vertices $s_1, \dots, s_k$ should not be considered entering any cluster of the branch decomposition containing s_i if it does not pick up its next delivery (or has picked up its last delivery) within the cluster of s_i. This twist demands a very careful design of the dynamic program by working with distances rather than explicit paths.

Then, our dynamic program works as follows: The algorithm computes the best solution at a given cluster C of the decomposition, for any prescribed number of tours (rounded to a power of $1+\tilde{\epsilon}$) entering and leaving C. This is done by iterating over all pairs of (pre-computed) solutions for the child clusters of C that are consistent with (namely, that potentially *can* lead to) the prescribed number of tours entering and leaving at C. Given consistent solutions for the child cluster, the optimal cost of combining them (given the constraints on the number of tours entering at C) is then computed through a min-cost assignment.

II. Related work

TSP in Euclidean and doubling metrics: Arora [5] and Mitchell [6] gave polynomial-time approximation schemes (PTASs) for TSP (Arora's algorithm is a PTAS for any fixed dimension). Rao and Smith [22] gave an $O(n\log n)$ approximation scheme for bounded-dimension Euclidean TSP, later improved to linear-time by Bartal and Gottlieb [7]. For TSP in doubling metrics, Talwar [10] gave a QPTAS; Bartal et al. [57] gave a PTAS; and Gotlieb [11] gave efficient PTAS.

TSP and subset TSP in minor-closed families: For TSP problem in planar graphs, Grigni et al. [3] gave the first (inefficient) PTAS for *unweighted* graphs; Arora et al. [4] extended Grigni et al. [4] to weighted graphs; Klein [8] designed the first EPTAS by introducing the contraction decomposition framework. Borradaile et al. [28] generalized Klein's EPTAS to bounded-genus graphs. The first PTAS for K_r-minor-free graph was designed by Demaine et al. [26] that improved upon the QPTAS by Grigni [12]. Recently, Borradaile et al. [13] obtained an EPTAS for TSP in K_r-minor-free graphs by connstructing light spanners; this work completed a long line of research on approximating classical TSP in K_r-minor-free graphs.

For subset TSP, Arora et al. [4] designed the first QPTAS for weighted planar graphs. Klein [21] obtained the first EPTAS for subset TSP in planar graphs by constructing a light planar subset spanner. Borradaile et al. [28] generalized Klein's subset spanner construction to bounded-genus graphs, thereby obtained an EPTAS. Le [14] designed the first (inefficient) PTAS for subset TSP in minor-free graphs. Our Theorem 5 completed this line of research.

[6] For simplicity, we work with branch decompositions

Light (subset) spanners: Light and sparse spanners were introduced for distributed computing [58, 59, 60]. Since then, spanners attract ever-growing interest; see [61] for a survey. Over the years, light spanners with constant lightness have been shown to exist in Euclidean metrics [22, 23], doubling metrics [11, 24], planar graphs [25], bounded genus graphs [12] and minor-free graphs [13]. For subset spanners, relevant results include subset spanners with constant lightness for planar graphs by Klein [21], for bounded genus graphs by Borradaile et al. [28]. Le [14] constructed subset spanners with lightness $O(\log |K|)$ for minor-free graphs.

Capacitated vehicle routing: There is a rich literature on the capacitated vehicle routing problem. When Q is arbitrary, the problem becomes extremely difficult; there is no known PTAS for any non-trivial metric. For $\mathbb{R}^2$, there is a QPTAS by Mathieu and Das for $\mathbb{R}^2$ [30] and for tree metrics, there is a (tight) $\frac{4}{3}$-approximation algorithm by Becker [31]. In general graphs, Haimovich and Rinnooy Kan [62] designed a 2.5-approximation algorithm.

In Euclidean spaces, better results were known for restricted values of Q: PTASes in $\mathbb{R}^2$ for $Q = O(2^{\log^{O_\epsilon(1)} n})$ by a sequence of papers [62, 2, 63] and for $Q = \Omega(n)$ by Asano et al. [2]; a PTAS in $\mathbb{R}^d$ for $Q = O(\log n^{1/d})$ by Khachay and Dubinin [64].

For constant Q, progress has been made on designing approximation schemes for various minor-closed families of graphs. Becker et al. [32] gave a QPTAS for planar and bounded-genus graphs. Later, Becker et al. [33] designed a PTAS for planar graphs.

Other relevant work includes a PTAS for graphs of bounded highway dimension and constant Q [42], a bicriteria PTAS for tree metrics and arbitrary Q [65], and an exact algorithm for treewidth-tw graphs with running time $O(n^{\text{tw}Q})$ [42].

III. Proof Overviews

A. Light subset spanners for minor-free metrics

In this section, we give a proof overview and review the main technical ideas for the proof of Theorem 1. A subgraph H of a graph G is called a *subset L-local* $(1+\epsilon)$-spanner of G with respect to a set K of terminals if $\forall t_1, t_2 \in K$ s.t. $d_G(t_1,t_2) \le L$, it holds that

$$d_H(t_1,t_2) \le (1+\epsilon)\cdot d_G(t_1,t_2)$$

Our starting point is the following reduction of Le [14].

Theorem 9 (Theorem 1.4 [14])**.** *Fix an $\epsilon \in (0,1)$. Suppose that for any K_r-minor-free weighted graph $G = (V,E,w)$, subset $K \subseteq V$ of terminals, and parameter $L > 0$, there is a subset L-local $(1+\epsilon)$-spanner w.r.t. K of weight at most $O_r(|K|\cdot L\cdot \text{poly}(\frac{1}{\epsilon}))$. For any terminal set, G admits a subset $(1+\epsilon)$-spanner with lightness $O_r(\text{poly}(\frac{1}{\epsilon}))$.*

Our main focus is to construct a light subset L-local spanner.

Proposition 1. *For any edge-weighted K_r-minor-free graph $G = (V,E,w)$, any subset $K \subseteq V$ of terminals, and any parameter $L > 0$, there is a subset L-local $(1+\epsilon)$-spanner for G with respect to K of weight $O_r(|K|\cdot L\cdot \text{poly}(\frac{1}{\epsilon}))$.*

Theorem 1 follows directly by combining Theorem 9 with Proposition 1. Our focus now is on proving Proposition 1. The proof is divided into two steps: in step 1 we solve the problem on the restricted case of planar graphs with bounded diameter and a single vortex. Then, in step 2, we reduce the problem from K_r-minor-free graphs to the special case solved in step 1.

Step 1: Single vortex with bounded diameter: The main lemma in step 1 is stated below. We define a *single-vortex* graph $G = G_\Sigma \cup W$ as a graph whose edge set can be partitioned into two parts G_Σ, W such that G_Σ induces a plane graph and W is a vortex of width [7] at most h glued to some face of G_Σ.

Lemma 1 (Single Vortex with Bounded Diameter)**.** *Consider a single-vortex graph $G = G_\Sigma \cup W$ with diameter $D = O_h(L)$, where G_Σ is planar, and W is a vortex of width at most h glued to a face of G_Σ. For any terminal set K, there exists a subset L-local $(1+\epsilon)$-spanner for G with respect to K of weight $O_h(|K|L\cdot\text{poly}(\frac{1}{\epsilon}))$.*

Let k be the number of terminals. The basic idea in constructing the spanner for Lemma 1 is to recursively break down the graph into a *hierarchy of clusters* where (1) the boundary of each cluster consists of a constant number of shortest paths, (2) each leaf cluster contains a constant number of terminals and (3) the number of clusters is $O(k)$. To break the graph, we use a variant of shortest path separators of Abraham and Gavoille [47]. Unlike general K_r-minor-free graphs, every shortest path in the separator of $G_\Sigma \cup W$ is a shortest path of the input graph, and hence, each has length at most L. In each recursive step, we use a shortest path separator to either reduce the number of terminals or reduce the number of paths in the boundary. For each cluster in the hierarchy, we add a *bipartite spanner* between every pair of shortest paths in the boundary of the cluster; a bipartite spanner is a set of edges that preserve all pairwise distances between two paths such that its weight is proportional to the distance between the paths and their lengths. Since the shortest paths have length $O(L)$ and we only preserve distances

[7] The *width* of the vortex is the width of its path decomposition.

of length at most L between terminals, the weight of the bipartite spanner is $O(L\text{poly}(\frac{1}{\epsilon}))$. That is, for each cluster in the recursive decomposition, we add weight $O(L\text{poly}(\frac{1}{\epsilon})$ and by property (3), the total weight of all bipartite spanners is $O(kL\text{poly}(\frac{1}{\epsilon})$ as desired.

Step 2: From minor-free to single vortex with bounded diameter.: We generalize the spanner construction of Step 1 to minor-free graphs using the Robertson-Seymour decomposition. We have five sub-steps, each generalizing further (at the expense of increasing the weight of the spanner by an additive term $O_h(kL \cdot \text{poly}(\frac{1}{\epsilon})))$.

Thus, consider the construction proposed in Step 1. In the first sub-step, we remove the assumption on the bounded diameter and make our spanner construction work for arbitrary planar graphs with a single vortex. The approach is as follows: Break a graph with unbounded diameter to overlapping clusters of diameter $O_h(L)$ such that every pair of vertices at distance at most L belongs to some cluster, and each vertex belong to at most $O_h(1)$ clusters. This is done using Abraham et al. sparse covers [51]. Then construct a spanner for each cluster separately by applying the approach of Step 1, namely Lemma 1, and return the union of these spanners. More concretely, we prove the following lemma.

Lemma 2 (Single Vortex)**.** *Consider a graph $G = G_\Sigma \cup W$ where G_Σ is planar, and W is a vortex of width at most h glued to a face of G_Σ. For any terminal set K, there exists a subset L-local $(1+\epsilon)$-spanner for G with respect to K of weight $O_h(|K|L \cdot \text{poly}(\frac{1}{\epsilon}))$.*

In the second sub-step, we generalize to planar graphs with at most h vortices of width [7] h. The basic idea is to "merge" all vortices into a single vortex of width $O(h^2)$. This is done by repeatedly deleting a shortest path between pairs of vortices, and "opening up" the cut to form a new face. The two vortices are then "merged" into a single vortex – in other words, they can be treated as a single vortex by the algorithm obtained at the first sub-step. This is repeated until all the vortices have been "merged" into a single vortex, at which point Lemma 2 applies. Here we face a quite important technical difficulty: when opening up a shortest path between two vortices, we may alter shortest paths between pairs of terminals (e.g.: the shortest path between two terminals intersects the shortest path between our two vortices, in which case deleting the shortest path between the vortices destroys the shortest path between the terminals). To resolve this issue, we compute a single-source spanner from each terminal to every nearby deleted path, thus controlling the distance between such terminal pairs in the resulting spanner. The above idea is captured in the following lemma.

Lemma 3 (Multiple Vortices)**.** *Consider a graph $G = G_\Sigma \cup W_1 \cup \cdots \cup W_{h'}$, where G_Σ is planar, $h' \leq h$, and each W_i is a vortex of width at most h glued to a face of G_Σ. For any terminal set K, there exists a subset L-local $(1+\epsilon)$-spanner for G with respect to K of weight $O_h(|K|L \cdot \text{poly}(\frac{1}{\epsilon}))$.*

In our third sub-step, we generalize to graphs of bounded genus with multiple vortices. The main tool here is "vortex paths" from [47]. Specifically, we can remove two vortex paths and reduce the genus by one (while increasing the number of vortices). Here each vortex path consists of essentially $O_h(1)$ shortest paths. We apply this genus reduction repeatedly until the graph has genus zero. The graph then has $O(g)$ new vortices. Next, we apply Lemma 3 to create a spanner. The technical difficulty of the previous step arises here as well: There may be shortest paths between pairs of terminals that intersect the vortex paths. We handle this issue in a similar manner.

Lemma 4 (Multiple Vortices and Genus)**.** *Consider a graph $G = G_\Sigma \cup W_1 \cup \cdots \cup W_{h'}$ where G_Σ is (cellularly) embedded on a surface Σ of genus at most $g = O(h)$, $h' \leq h$, and each W_i is a vortex of width at most h glued to a face of G_Σ. For any terminal set K, there exists a subset L-local $(1+\epsilon)$-spanner for G with respect to K of weight $O_h(|K|L \cdot \text{poly}(\frac{1}{\epsilon}))$.*

In our fourth sub-step, we generalize to nearly h-embeddable graphs. That is, in addition to genus and vortices, we also allow G to have at most h apices. The spanner is constructed by first deleting all the apices and applying Lemma 4. Then, in order to compensate for the deleted apices, we add a shortest path from each apex to every terminal at distance at most L.

Lemma 5 (Nearly h-Embeddable)**.** *Consider a nearly h-embeddable graph G with a set K of k terminals. There exists an L-local $(1+\epsilon)$-spanner for K of weight $O_h(kL \cdot \text{poly}(\frac{1}{\epsilon}))$.*

Finally, in our last sub-step, we generalize to minor-free graphs, thus proving Proposition 1. Recall that according to [15] a minor graph can be decomposed into a clique-sum decomposition, where each node in the decomposition is nearly h-embeddable. Our major step here is transforming the graph G into a graph G' that preserves all terminal distances in G, while having at most $O(k)$ bags in its clique-sum decomposition. This is done by first removing leaf nodes which are not "essential" for any terminal distance, and then shrinking long paths in the decomposition where all internal nodes have degree two and (roughly) do not contain terminals. Next, given G', we make each vertex that belongs to one of the cliques in the clique-sum decomposition into a terminal. The new number of terminals is bounded by $O_h(k)$. The last step is simply to construct an internal spanner for each bag separately using Lemma 5, and return the union of the constructed spanners.

B. *Embedding into low-treewidth graphs*

At a high level, we follow the same approach as for the subset spanner. Due to the different nature of the constructed structures, and the different distortion guarantees, there are some differences that raise significant challenges.

To prove Theorem 4, which addresses bounded-genus graphs, we generalize the result of Fox-Epstein et al. [41]. Our approach is basically the same as for the subset spanner: we decompose the graph into simpler and simpler pieces by removing shortest paths. Here, instead of deleting a path, we will use a *cutting lemma*. However, in this setting it is not clear how to use single-source or bipartite spanners to compensate for the changes to the shortest-path metric due to path deletions, since these spanners may have large treewidth. Instead, we will *portalize* the cut path. That is, we add an ϵD-net[8] of the path to every bag of the tree decomposition of the host graph. Clearly, this strategy has to be used cautiously since it immediately increases the treewidth significantly.

Next we turn to the proof of Theorem 2, which addresses minor-free graphs. Here we again use the RS framework. Apices pose an interesting challenge. Standard techniques to deal with apices consist in removing them from the graph, solve the problem on the remaining graph which is planar, and add back the apices later [43, 44]. However, in our setting, removing apices can make the diameter of the resulting graph, say G', become arbitrarily larger than D and thus, it seems hopeless to embed G' into a low treewidth graph with an additive distortion bounded by D. This is where randomness comes into play: we use padded decomposition [19] to randomly partition G' into pieces of (strong) diameter $D' = O(\frac{D}{\epsilon})$. We then embed each part of the partition (which is planar) separately into graphs of bounded treewidth with additive distortion $\epsilon^2 D' = O(\epsilon D)$, add back the apices by connecting them to all the vertices of all the bounded treewidth graphs (and so adding all of them to each bag of each decomposition) and obtain a graph with bounded treewidth and an *expected* additive distortion ϵD.

Our next stop is to find bounded treewidth embeddings of clique-sums of bounded genus graphs with apices. Suppose that G is decomposed into clique-sums of graphs $G_1, G_2, \dots, G_k$. We call each G_i a *piece*. A natural idea is to embed each G_i into a low-treewidth graph H_i, called the *host graph* with a tree decomposition $\mathcal{T}_i$, and then combine all the tree decompositions together. Suppose that G_1 and G_2 participate in the clique-sum decomposition of G using the clique Q. To merge G_1 and G_2, we wish to have an embedding from G_i to H_i, $i = 1, 2$, that *preserves* the clique Q in the clique-sum of G_1 and G_2. That is, the set of vertices $\{f_i(v) | v \in Q\}$ induces a clique in H_i (so that there will be bag in the tree decomposition of H_i containing $f(Q)$). However, it is impossible to have such an embedding even if all G_i's are planar.[9] To overcome this obstacle, we will allow each vertex in G_i to have multiple images in H_i. Specifically, we introduce the notion of*one-to-many* embeddings. Note that given a one-to-many embedding, one can construct a classic embedding by identifying each vertex with an arbitrary copy.

Definition 1 (One-to-many embedding). *An embedding $f : G \to 2^H$ of a graph G into a graph H is a* one-to-many embedding *if for every $v \in G$, $f(v)$ is a non empty set of vertices in H, where the sets $\{f(v)\}_{v \in G}$ are disjoint.*

We say that f is dominating *if for every pair of vertices $u, v \in G$, it holds that $d_G(u, v) \le \min_{u' \in f(u), v' \in f(v)} d_H(u', v')$. We say that f has additive distortion ϵD if it is dominating and $\forall u, v \in G$ it holds that $\max_{u' \in f(u), v' \in f(v)} d_H(u', v') \le d_G(u, v) + \epsilon D$. Note that, as for every vertex $v \in G$, $d_G(v, v) = 0$, having additive distortion ϵD implies that all the copies in $f(v)$ are at distance at most ϵD from each other. The method of Fox-Epstein et al. [41] yielded a one-to-many embedding but this aspect of the embedding was not important to their result. Here we use to address the clique-preservation problem discussed above.*

A stochastic one-to-many embedding is a distribution $\mathcal{D}$ over dominating one-to-many embeddings. We say that a stochastic one-to-many embedding has expected additive distortion ϵD if $\forall u, v \in G$ it holds that $\mathbb{E}[\max_{u' \in f(u), v' \in f(v)} d_H(u', v')] \le d_G(u, v) + \epsilon D$.

We can show that in order to combine the different one-to-many embeddings of the pieces $G_1, \dots, G_s$, it is enough that for every clique Q we will have a bag B containing at least one copy of each vertex in Q. Formally,

[8] An r-net of a set A, is a set $N \subset A$ of vertices all at distance at least r from each other, and such that every $v \in A$ has a net point $t \in N$ at distance at most r. If A is a path of length L, then for every r-net N, $|N| = O(\frac{L}{r})$.

[9] To see this, suppose that G is clique-sums of a graph G_0 with many other graphs $G_1, G_2 \dots, G_s$ in a star-like way, where G_0 has treewidth polynomial in n, and every edge of G_0 is used for some clique sum. If H_0 preserves all cliques, it contains G_0 and thus has treewidth polynomial in n.

Definition 2 (Clique-preserving embedding). *A one-to-many embedding $f : G \to 2^H$ is called clique-preserving embedding if for every clique $Q \in G$, there is a clique Q' in H such that for every vertex $v \in Q$, $f(v) \cap Q' \neq \emptyset$.*

While it is impossible to preserve all cliques in a one-to-one embedding, it is possible to preserve all cliques in a one-to-many embedding; this is one of our major conceptual contributions. One might worry about the number of maximal cliques in G. However, since G has constant degeneracy, the number of maximal cliques is linear [52]. Suppose that f is clique-preserving, and let $\mathcal{T}$ be some tree-decomposition of H. Then for every clique Q in G, there is a bag of $\mathcal{T}$ containing a copy of (the image of) Q in H.

We now have the required definitions, and begin the description of the different steps in creating the embedding. The most basic case we are dealing with directly is that of a planar graph with a single vortex and diameter D into a graph of treewidth $O(\frac{\log n}{\epsilon})$ and additive distortion ϵD. The high level idea, similarly to our subset spanner, is to use vortex-path separator to create a hierarchical partition tree τ. The depth of the tree will be $O(\log n)$. To accommodate for the damage caused by the separation, we *portalize* each vortex-path in the separator. That is for each such path Q, we pick an ϵD-net [8] N_Q of size $O(\frac{1}{\epsilon})$. The vertices of N_Q called *portals*. Since each node of τ is associated with a constant number of vortex-paths, there are at most $O(\frac{1}{\epsilon})$ portals corresponding to each node of τ. Thus, if we collect all portals along the path from a leaf to the root of τ, there are $O(\frac{\log n}{\epsilon})$ portals. We create a bag for each leaf Υ of the tree τ. In addition for each bag we add the portals corresponding to nodes along the path from the root to Υ. The tree decomposition is then created w.r.t. τ. Finally, we need to make the embedding clique-preserving. Consider a clique Q, there will be a leaf Υ_Q of τ containing a sub-clique $Q' \subseteq Q$, while all the vertices in $Q \setminus Q'$ belong to paths in the boundary of Υ_Q. We will create a new bag containing (copies) of all the vertices in Q and all the corresponding portals. The vertices of Q' will have a single copy in the embedding, while the distortion of the vertices $Q' \subseteq Q$ will be guaranteed using a nearby portal.

Lemma 6 (Single Vortex with Bounded Diameter). *Given a single-vortex graph $G = G_\Sigma \cup W$ where the vortex W has width h. There is a one-to-many, clique-preserving embedding f from G to a graph H with treewidth $O(\frac{h \log n}{\epsilon})$ and additive distortion ϵD where D is diameter of G.*

We then can extend the embedding to planar graphs with multiple vortices using the *vortex merging* technique, and then to graphs embedded on a genus-g surface with multiple vortices by cutting along vortex-paths. The main tool here is a *cutting lemma* which bound the diameter blowup after each cutting step. At this point, the embedding is still deterministic.

Lemma 7 (Multiple Vortices). *Consider a graph $G = G_\Sigma \cup W_1 \cup \cdots \cup W_{v(G)}$ of diameter D, where G_Σ can be drawn on the plane, and each W_i is a vortex of width at most h glued to a face of G_Σ, and $v(G)$ is the number of vortices in G. There is a one-to-many, clique-preserving embedding f from G to a graph H of treewidth at most $\frac{h 2^{O(v(G))} \log n}{\epsilon}$ with additive distortion ϵD.*

Lemma 8 (Multiple Vortices and Genus). *Consider a graph $G = G_\Sigma \cup W_1 \cup \cdots \cup W_{v(G)}$ of diameter D, where G_Σ is (cellularly) embedded on a surface Σ of genus $g(G)$, and each W_i is a vortex of width at most h glued to a face of G_Σ. There is a one-to-many clique-preserving embedding f from G to a graph H of treewidth at most $\frac{h 2^{O(v(G) g(G))} \log n}{\epsilon}$ with additive distortion ϵD.*

We then extend the embedding to graphs embedded on a genus-g surface with multiple vortices and apices (a.k.a. nearly embeddable graphs). The problem with apices, as pointed out at the beginning of this section, is that the diameter of the graph after removing apices could be unbounded in terms of the diameter of the original graph. Indeed, while the embedding in Lemma 8 is deterministic, it is not clear how to *deterministically* embed a nearly embeddable graph into a bounded treewidth graph with additive distortion ϵD. We use padded decompositions [19] to decompose the graph into clusters of strong diameter $O(D/\epsilon)$, embed each part separately, and then combine all the embeddings into a single graph. Note that separated nodes will have additive distortion as large as $2D$, however, this will happen with probability at most $O(\epsilon)$. To make this embedding clique-preserving, we add to each cluster its neighborhood. Thus some small fraction of the vertices will belong to multiple clusters. As a result, we obtain a one-to-many stochastic embedding with expected additive distortion ϵD.

Lemma 9 (Nearly h-Embeddable). *Given a nearly h-embeddable graph G of diameter D, there is a one-to-many stochastic clique-preserving embedding into graphs with treewidth $O_h(\frac{\log n}{\epsilon^2})$ and expected additive distortion ϵD. Furthermore, every bag of the tree decomposition of every graph in the support contains (the image of) the apex set of G.*

Finally we are in the case of general minor free graph $G = G_1 \oplus_h G_2 \oplus_h \ldots \oplus_h G_s$. We sample an embedding for each G_i using Lemma 9 to some bounded treewidth graph H_i. As all these embeddings are clique-preserving, there is a natural way to combine the tree decompositions of all the graphs H_i together. Here we run into another challenge: we need to guarantee that the additive distortion caused by merging tree decompositions is not too large. To explore this challenge, let us consider the clique-sum decomposition tree $\mathcal{T}$ of G: each node of $\mathcal{T}$ corresponds uniquely to G_i for some i, and that G is obtained by clique-summing all adjacent graphs G_i and G_j in $\mathcal{T}$. Suppose that $\mathcal{T}$ has a (polynomially) long path $\mathcal{P}$ with hop-length p. Then, for a vertex u in the graph corresponding to one end of $\mathcal{P}$ and a vertex v in the graph corresponding to another end of $\mathcal{P}$, the additive distortion between u and v could potentially $p\epsilon D$ since every time the shortest path between u and v goes through a graph G_i, we must pay additive distortion ϵD in the embedding of G_i. When p is polynomially large, the additive distortion is polynomial in n. We resolve this issue by the following idea:(1) pick a *separator piece* G_i of $\mathcal{T}$ (G_i is a separator of $\mathcal{T}$ if each component $\mathcal{T} \setminus G_i$ has at most $2/3$ the number of pieces of $\mathcal{T}$), (2) recursively embed pieces in subtrees of $\mathcal{T} \setminus G_i$ and (3) add the join set between G_i and each subtree, say $\mathcal{T}'$ of $\mathcal{T} \setminus G_i$ to all bags of the tree decomposition corresponding to $\mathcal{T}'$. We then can show that this construction incurs another additive $\log n$ factor in the treewidth while insuring a total additive distortion of ϵD. Hence the final tree decomposition has width $O(\frac{\log n}{\epsilon^2})$.

An interesting consequence of our one-to-many embedding approach is that the host graphs H will contain Steiner points. That is, its vertex set will be greater than V. We do not know whether it is possible to obtain the properties of Theorem 2 while embedding into n-vertex graphs. In this context, the Steiner point removal problem studies whether it is possible to remove all Steiner points while preserving both pairwise distance and topological structure [53, 54]. Unfortunately, in general, even if G is a tree, a multiplicative distortion of 8 is necessary [55]. Nevertheless, as Krauthgamer et al. [56] proved, given a set K of k terminals in a graph H of treewidth tw, we can embed the terminal set K isometrically (that is with multiplicative distortion 1) into a graph with $O(k \cdot \mathrm{tw}^3)$ vertices and treewidth tw. It follows that we can ensure that all embeddings in the support of the stochastic embedding in Theorem 2 are into graphs with $O_r(n \cdot \frac{\log^3 n}{\epsilon^6})$ vertices.

References

[1] C. H. Papadimitriou and M. Yannakakis, "The traveling salesman problem with distances one and two," *Mathematics of Operations Research*, vol. 18, no. 1, pp. 1–11, 1993. 1

[2] T. Asano, N. Katoh, H. Tamaki, and T. Tokuyama, "Covering points in the plane by k-tours: Towards a polynomial time approximation scheme for general k," in *29th Annual ACM Symposium on Theory of Computing*, 1997, p. 275–283. 1, 2, 6

[3] M. Grigni, E. Koutsoupias, and C. H. Papadimitriou, "An approximation scheme for planar graph TSP," in *36th Annual Symposium on Foundations of Computer Science*, 1995, pp. 640–645. 1, 5

[4] S. Arora, M. Grigni, D. R. Karger, P. N. Klein, and A. Woloszyn, "A polynomial-time approximation scheme for weighted planar graph TSP," in *9th Annual ACM-SIAM Symposium on Discrete Algorithms, 25-27 January 1998, San Francisco, California, USA*, 1998, pp. 33–41. 1, 4, 5

[5] S. Arora, "Nearly linear time approximation schemes for euclidean TSP and other geometric problems," in *38th Annual Symposium on Foundations of Computer Science*, 1997, pp. 554–563. 1, 5

[6] J. 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.*, vol. 28, no. 4, pp. 1298–1309, 1999. 1, 5

[7] Y. Bartal and L. Gottlieb, "A linear time approximation scheme for euclidean TSP," in *54th Annual IEEE Symposium on Foundations of Computer Science*, 2013, pp. 698–706. 1, 5

[8] P. N. Klein, "A linear-time approximation scheme for planar weighted TSP," in *46th Annual IEEE Symposium on Foundations of Computer Science*, 2005, pp. 647–657. 1, 2, 5

[9] ——, "A linear-time approximation scheme for TSP in undirected planar graphs with edge-weights," *SIAM J. Comput.*, vol. 37, no. 6, pp. 1926–1952, 2008. 1

[10] K. Talwar, "Bypassing the embedding: algorithms for low dimensional metrics," in *36th Annual ACM Symposium on Theory of Computing*, 2004, pp. 281–290. 1, 5

[11] L. Gottlieb, "A light metric spanner," in *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, 2015, pp. 759–772. 1, 2, 5, 6

[12] M. Grigni, "Approximate TSP in graphs with forbidden minors," in *Automata, Languages and Programming, 27th International Colloquium, ICALP 2000, Geneva, Switzerland, July 9-15, 2000, Proceedings*, 2000, pp. 869–877. 1, 5, 6

[13] G. Borradaile, H. Le, and C. Wulff-Nilsen, "Minor-free graphs have light spanners," in *2017 IEEE 58th Annual Symposium on Foundations of Computer Science*, ser. FOCS '17, 2017, pp. 767–778. 1, 2, 5, 6

[14] H. Le, "A PTAS for subset TSP in minor-free graphs," in *31st ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, 2020, pp. 2279–2298, full version: arxiv:1804.01588. 2, 4, 5, 6

[15] N. Robertson and P. D. Seymour, "Graph minors. XVI. Excluding a non-planar graph," *Journal of Combinatoral Theory Series B*, vol. 89, no. 1, pp. 43–76, 2003. 2, 4, 7

[16] P. N. Klein, S. A. Plotkin, and S. Rao, "Excluded minors, network decomposition, and multicommodity flow," in *Proceedings of the 25th Annual ACM Symposium on Theory of Computing*, ser. STOC '93, 1993, p. 682–690. 2

[17] I. Abraham, C. Gavoille, A. Gupta, O. Neiman, and K. Talwar, “Cops, robbers, and threatening skeletons: Padded decomposition for minor-free graphs,” *SIAM J. Comput.*, vol. 48, no. 3, pp. 1120–1145, 2019. 2

[18] J. Fakcharoenphol and K. Talwar, “An improved decomposition theorem for graphs excluding a fixed minor,” in *International Workshop on Approximation Algorithms for Combinatorial Optimization*, ser. APPROX ‘03, 2003, pp. 36–46. 2

[19] A. Filtser, “On strong diameter padded decompositions,” in *APPROX*, 2019, pp. 6:1–6:21. 2, 8, 9

[20] G. Borradaile, P. N. Klein, and C. Mathieu, “An $O(n \log n)$ approximation scheme for Steiner tree in planar graphs,” *ACM Transactions on Algorithms (TALG)*, vol. 5, no. 3, pp. 31:1–31:31, 2009. 2

[21] P. N. Klein, “Subset spanner for planar graphs, with application to subset TSP,” in *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, ser. STOC ’06, 2006, pp. 749–756. 2, 4, 5, 6

[22] S. B. Rao and W. D. Smith, “Approximating geometrical graphs via “spanners” and “banyans”,” in *Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, ser. STOC ’98, 1998, pp. 540–550, full version at http://graphics.stanford.edu/courses/cs468-06-winter/Papers/rs-tsp.pdf. 2, 5, 6

[23] H. Le and S. Solomon, “Truly optimal euclidean spanners,” in *60th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2019, Baltimore, Maryland, USA, November 9-12, 2019*, 2019, pp. 1078–1100. 2, 6

[24] G. Borradaile, H. Le, and C. Wulff-Nilsen, “Greedy spanners are optimal in doubling metrics,” in *Proceedings of the 30th Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA ‘19, 2019, pp. 2371–2379. 2, 6

[25] I. Althöfer, G. Das, D. Dobkin, D. Joseph, and J. Soares, “On sparse spanners of weighted graphs,” *Discrete Computational Geometry*, vol. 9, no. 1, pp. 81–100, 1993. 2, 6

[26] E. D. Demaine, M. Hajiaghayi, and K. Kawarabayashi, “Contraction decomposition in h-minor-free graphs and algorithmic applications,” in *43rd ACM Symposium on Theory of Computing*, 2011, pp. 441–450. 2, 5

[27] A. Filtser and S. Solomon, “The greedy spanner is existentially optimal,” in *Proceedings of the 2016 ACM Symposium on Principles of Distributed Computing*, ser. PODC ’16, 2016, pp. 9–17, to appear in SICOMP 2020. 2

[28] G. Borradaile, E. D. Demaine, and S. Tazari, “Polynomial-time approximation schemes for subset-connectivity problems in bounded-genus graphs,” *Algorithmica*, vol. 68, no. 2, pp. 287–311, 2014. 2, 5, 6

[29] G. B. Dantzig and J. H. Ramser, “The truck dispatching problem,” *Management science*, vol. 6, no. 1, pp. 80–91, 1959. 2

[30] A. Das and C. Mathieu, “A quasipolynomial time approximation scheme for euclidean capacitated vehicle routing,” *Algorithmica*, vol. 73, no. 1, p. 115–142, 2015. 2, 6

[31] A. Becker, “A tight 4/3 approximation for capacitated vehicle routing in trees,” in *APPROX*, 118, Ed., 2018, pp. 3:1–3:15. 6

[32] A. Becker, P. N. Klein, and D. Saulpic, “A quasi-polynomial-time approximation scheme for vehicle routing on planar and bounded-genus graphs,” in *25th Annual European Symposium on Algorithms*, ser. ESA ‘17, vol. 87, 2017, pp. 12:1–12:15. 2, 6

[33] A. Becker, P. N. Klein, and A. Schild, “A PTAS for bounded-capacity vehicle routing in planar graphs,” in *16th Workshop on Algorithms and Data Structures*, 2019, pp. 99–111. 2, 6

[34] D. Eisenstat, P. N. Klein, and C. Mathieu, “Approximating k-center in planar graphs,” in *25th Annual ACM-SIAM Symposium on Discrete Algorithms*, 2014, pp. 617–627. 3, 4

[35] Y. Bartal, “Probabilistic approximations of metric spaces and its algorithmic applications,” in *37th Annual Symposium on Foundations of Computer Science, FOCS ’96, Burlington, Vermont, USA, 14-16 October, 1996*, 1996, pp. 184–193. 3

[36] J. Fakcharoenphol, S. Rao, and K. Talwar, “A tight bound on approximating arbitrary metrics by tree metrics,” *J. Comput. Syst. Sci.*, vol. 69, no. 3, pp. 485–497, 2004. 3

[37] A. Chakrabarti, A. Jaffe, J. R. Lee, and J. Vincent, “Embeddings of topological graphs: Lossy invariants, linearization, and 2-sums,” in *49th Annual IEEE Symposium on Foundations of Computer Science*, 2008, pp. 761–770. 3

[38] S. Rao, “Small distortion and volume preserving embeddings for planar and Euclidean metrics,” in *15th Annual Symposium on Computational Geometry*, 1999, pp. 300–306. 3

[39] R. Krauthgamer, J. R. Lee, M. Mendel, and A. Naor, “Measured descent: a new embedding method for finite metrics,” *Geometric and Functional Analysis*, vol. 15, no. 4, pp. 839–858, 2005. 3

[40] I. Abraham, A. Filtser, A. Gupta, and O. Neiman, “Metric embedding via shortest path decompositions,” in *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, 2018, pp. 952–963, full version: . 3

[41] E. Fox-Epstein, P. N. Klein, and A. Schild, “Embedding planar graphs into low-treewidth graphs with applications to efficient approximation schemes for metric problems,” in *Proceedings of the 30th Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA ‘19, 2019, p. 1069–1088. 3, 8

[42] A. Becker, P. N. Klein, and D. Saulpic, “Polynomial-time approximation schemes for k-center, k-median, and capacitated vehicle routing in bounded highway dimension,” in *26th Annual European Symposium on Algorithms*, ser. ESA ‘18, vol. 112, 2018, pp. 8:1–8:15. 4, 6

[43] M. Grohe, “Local tree-width, excluded minors, and approximation algorithms,” *Combinatorica*, vol. 23, no. 4, pp. 613–632, 2003. 4, 8

[44] E. D. Demaine, M. Hajiaghayi, and K. Kawarabayashi, “Algorithmic graph minor theory: Decomposition, approximation, and coloring,” in *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, 2005, pp. 637–646. 4, 8

[45] G. Borradaile, C. Kenyon-Mathieu, and P. N. Klein, “A polynomial-time approximation scheme for Steiner tree in planar graphs,” in *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA ‘07, 2007, pp. 1285–1294. 4

[46] G. Borradaile and P. N. Klein, "The two-edge connectivity survivable network problem in planar graphs," in *35th International Colloquium on Automata, Languages and Programming*, 2008, pp. 485–501. 4

[47] I. Abraham and C. Gavoille, "Object location using path separators," in *25*. 4, 6, 7

[48] M. Thorup, "Compact oracles for reachability and approximate distances in planar digraphs," *J. ACM*, vol. 51, no. 6, pp. 993–1024, Nov. 2004. 4

[49] K. Kawarabayashi, P. N. Klein, and C. Sommer, "Linear-space approximate distance oracles for planar, bounded-genus and minor-free graphs," in *The 38th International Colloquium on Automata, Languages and Programming*, ser. ICALP '11, 2011, pp. 135–146. 4

[50] M. Thorup, "Compact oracles for reachability and approximate distances in planar digraphs," *Journal of the ACM*, vol. 51, no. 6, p. 993–1024, 2004. 4

[51] I. Abraham, C. Gavoille, D. Malkhi, and U. Wieder, "Strong-diameter decompositions of minor free graphs," *Theory of Computing Systems*, vol. 47, no. 4, pp. 837–855, 2010. 4, 7

[52] D. Eppstein, M. Löffler, and D. Strash, "Listing all maximal cliques in sparse graphs in near-optimal time," in *International Symposium on Algorithms and Computation*, 2010, pp. 403–414. 9

[53] A. Filtser, "Steiner point removal with distortion $O(logk)$ using the relaxed-voronoi algorithm," *SIAM J. Comput.*, vol. 48, no. 2, pp. 249–278, 2019. 10

[54] ——, "Scattering and sparse partitions, and their applications," *CoRR*, vol. abs/2001.04447, 2020, to appear in ICALP 2020. 10

[55] T.-H. Chan, D. Xia, G. Konjevod, and A. Richa, "A tight lower bound for the Steiner point removal problem on trees," in *Proceedings of the 9th International Conference on Approximation Algorithms for Combinatorial Optimization Problems, and 10th International Conference on Randomization and Computation*, ser. APPROX'06/RANDOM'06. Berlin, Heidelberg: Springer-Verlag, 2006, pp. 70–81. 10

[56] R. Krauthgamer, H. L. Nguyen, and T. Zondiner, "Preserving terminal distances using minors," *SIAM J. Discrete Math.*, vol. 28, no. 1, pp. 127–141, 2014. 10

[57] Y. Bartal, L. Gottlieb, and R. Krauthgamer, "The traveling salesman problem: Low-dimensionality implies a polynomial time approximation scheme," *SIAM J. Comput.*, vol. 45, no. 4, pp. 1563–1581, 2016. 5

[58] B. Awerbuch, "Communication-time trade-offs in network synchronization," in *4th Annual ACM Symposium on Principles of Distributed Computing*, 1985, pp. 272–276. 6

[59] D. Peleg and A. A. Schäffer, "Graph spanners," *Journal of Graph Theory*, vol. 13, no. 1, pp. 99–116, 1989. 6

[60] B. Awerbuch, A. Baratz, and D. Peleg, "Efficient broadcast and light-weight spanners," *Technical Report CS92-22, Weizmann Institute*, October, 1992. 6

[61] R. Ahmed, G. Bodwin, F. D. Sahneh, K. Hamm, M. J. L. Jebelli, S. Kobourov, and R. Spence, "Graph spanners: A tutorial review," *arXiv preprint arXiv:1909.03152*, 2019, . 6

[62] M. Haimovich and A. H. G. R. Kan, "Bounds and heuristics for capacitated routing problems," *Mathematics of Operations Research*, vol. 10, no. 4, pp. 527–542, 1985. 6

[63] A. Adamaszek, A. Czumaj, and A. Lingas, "PTAS for k-tour cover problem on the plane for moderately large values of k," in *20th International Symposium on Algorithms and Computation*, 2009, pp. 994–1003. 6

[64] M. Khachay and R. Dubinin, "PTAS for the Euclidean capacitated vehicle routing problem in $\mathbb{R}^d$," in *Discrete Optimization and Operations Research*, 2016, pp. 193–205. 6

[65] A. Becker and A. Paul, "A framework for vehicle routing approximation schemes in trees," in *16th Workshop on Algorithms and Data Structures*, 2019, pp. 112–125. 6