Breaching the 2-Approximation Barrier for Euclidean Capacitated Vehicle Routing

Zachary Friggstad* Fabrizio Grandoni† Ramin Mousavi‡

Abstract. In the (Unit Demand) Euclidean Capacitated Vehicle Routing problem (CVRP), we are given a collection of n points in the Euclidean plane (the *clients*), one extra point (the *depot*), and one integer $Q \geq 1$ (the *vehicle capacity*). A feasible solution is a collection of tours, where each tour contains the depot and at most Q clients, such that each client belongs to at least one such tour. Our goal is to minimize the total length of the tours. This models, e.g., the problem of delivering identical items stored at the depot to clients using a single vehicle that can carry at most Q items at a time.

CVRP is NP-hard (since it generalized the Euclidean TSP problem) and well-studied in terms of approximation algorithms. The current best (polynomial-time) approximation algorithm for CVRP works as follows. First one computes a (TSP) tour T_{tsp} visiting all the clients and the depot using the PTAS for Euclidean TSP by [Arora, '98; Mitchell, '99]. Notice that T_{tsp} has length at most $1+\varepsilon$ times the optimal CVRP cost opt, for an arbitrarily small constant $\varepsilon > 0$. Then one splits T_{tsp} into tours containing at most Q clients each, using a simple tour splitting heuristic by [Haimovich and Kan, '85]: this step increases the cost of the solution by the so-called *radial lower bound* $\mathrm{rlb} \leq \mathrm{opt}$. Altogether this gives a $2+\varepsilon$ approximation.

In this paper we break the long-standing 2-approximation barrier for CVRP, by presenting a $2-\tau$ approximation for a small positive constant $\tau > 10^{-5}$. Let $\mathrm{rlb} = (1-\gamma)\mathrm{opt}$ for some $\gamma \in [0,1]$. The above algorithm provides a $2+\varepsilon-\gamma$ approximation, hence bad when the radial lower bound is large. We complement this with a new and drastically different algorithm, based on a careful use of matroid intersection, which achieves a $1+\varepsilon+O(\sqrt[5]{\gamma})$ approximation. Hence the approximation factor of our new algorithm approaches 1 when rlb approaches opt.

1 Introduction. In the *(Unit Demand) Euclidean Capacitated Vehicle Routing* problem (CVRP) we are given an integer $Q \geq 1$ (the *vehicle capacity*) plus $n+1$ points $V = \mathcal{C} \cup \{r\}$ in the Euclidean plane, where r is the *depot* and $\mathcal{C}$ are the *clients*. We let $c(a,b) = c(b,a)$ denote the Euclidean distance between any two points a, b in the Euclidean plane. Notice that we allow $a, b \in V$ to be collocated, in which case $c(a,b) = 0$. It is also convenient to interpret the input points as a complete undirected edge-weighted graph $G = (V, E, c)$, where the cost of edge $\{a,b\} \in E$ is naturally $c(\{a,b\}) = c(a,b)$. Given a subgraph S of G, we let $V(S)$ (resp., $E(S)$) denote its nodes (resp., edges). We also let $\mathcal{C}(S) := V(S) \setminus \{r\}$ be the clients in S. A feasible solution is a collection of tours $T_1, \ldots, T_q$ such that: (1) each tour T_i contains the depot and at most Q other clients (i.e., $r \in V(T_i)$ and $|\mathcal{C}(T_i)| \leq Q$) and (2) each client is contained in at least one tour T_i, i.e, $\cup_{i=1}^{q} \mathcal{C}(T_i) = \mathcal{C}$. Our goal is to compute a feasible solution $\mathrm{OPT} = (T_1^*, \ldots, T_\Lambda^*)$ of minimum cost $\mathrm{opt} := c(\mathrm{OPT}) := \sum_{i=1}^{\Lambda} c(T_i^*)$, where $c(T_i^*) := \sum_{e \in E(T_i^*)} c(e)$ is the length of the i-th tour. As an application, the reader might think about a delivery service which owns a collection of (at least n) identical items located in a depot, and has to deliver one item to each one of n client. In order to do that, it exploits a vehicle that can carry at most Q items at a time. The goal is to minimize the distance traveled by the vehicle.

CVRP and its generalizations are among the most fundamental and best studied routing problems, and their study dates back as far as 1959 by Dantzig and Ramser [11]. There are also books dedicated to this topic, see, e.g., [28].

*Department of Computing Science, University of Alberta (zacharyf@ualberta.ca). Research supported by an NSERC Discovery Grant

†IDSIA, USI-SUPSI (farbizio.grandoni@gmail.com). Partially supported by the SNF Grants 200021_200731 and 200021-236706

‡IDSIA, USI-SUPSI (ram.mousavi@gmail.com). Partially supported by the SNF Grant 200021_200731

CVRP includes the Euclidean TSP problem as a special case (when $Q \geq n$, hence one tour is sufficient), and it is therefore NP-hard [24]. This motivated its study in terms of approximation algorithms. The current best (polynomial-time) approximation factor for CVRP is $2+\varepsilon$ for any constant $\varepsilon > 0$. For the later discussion, it is instructive to see how it works. First of all, one computes a TSP tour T_{tsp} visiting all the clients and the depot. Let tsp be the length of a shortest such tour. Trivially tsp $\leq$ opt: indeed, by gluing together the tours in OPT and shortcutting the copies of r by means of triangle inequality, one gets a feasible TSP tour of $V \cup \{r\}$. Therefore, using the PTAS for Euclidean TSP by Arora [3] or Mitchell [21], one obtains $c(T_{tsp}) := \sum_{e \in E(T_{tsp})} c(e) \leq (1+\varepsilon) \cdot \mathrm{tsp} \leq (1+\varepsilon) \cdot \mathrm{opt}$ for an arbitrarily small constant $\varepsilon > 0$. Next one converts T_{tsp} into a collection of feasible tours using the *tour splitting heuristic* by Haimovich and Kan [15]. In more detail, let $r, v_1, \ldots, v_n, r$ be the ordered sequence of nodes in T_{tsp}. Choose a random[1] integer offset $s \in \{1, \ldots, Q\}$. The collection of tours is $T_0 = r, v_1, \ldots, v_s, r$, $T_1 = r, v_{s+1}, \ldots, v_{s+Q}, r$, $T_2 = r, v_{s+Q+1}, \ldots, v_{s+2 \cdot Q}, r$, $\ldots$, $T_{h-1} = r, v_{s+(h-1) \cdot Q+1}, \ldots, v_n, r$. Observe that these tours contain at most Q clients each and together they cover all the clients (hence they induce a feasible solution). The increase of the cost w.r.t. $c(T_{tsp})$ is upper bounded by $\sum_{i=0}^{h-1} 2 \cdot c(r, v_{s+i \cdot Q})$. To upper bound the latter increase, one can use the following *radial lower bound*:

Lemma 1.1 (Haimovich and Kan [15]). *One has* $\mathrm{rlb} := \frac{1}{Q} \sum_{v \in \mathcal{C}} 2 \cdot c(r, v) \leq \mathrm{opt}$.

Proof. For each tour $T^* \in$ OPT and each client $v \in \mathcal{C}(T^*)$, we have $2 \cdot c(r,v) \leq c(T^*)$ because T^* consists of two disjoint r-v paths. So $\sum\limits_{v \in \mathcal{C}(T^*)} 2 \cdot c(r,v) \leq |\mathcal{C}(T^*)| \cdot c(T^*) \leq Q \cdot c(T^*)$. Summing over all tours $T^* \in$ OPT shows $\sum_{v \in \mathcal{C}} 2 \cdot c(r,v) \leq Q \cdot \mathrm{opt}$, as required. ∎

The expected contribution of each node v to the increase of the cost is at most $\frac{2}{Q} \cdot c(r,v)$. Hence the total expected increase is at most rlb. Altogether, the expected cost of the computed solution is at most $(1+\varepsilon) \cdot \mathrm{tsp} + \mathrm{rlb} \leq (2+\varepsilon) \cdot \mathrm{opt}$.

There are a number of improved approximation algorithms for special cases of CVRP. Asano et al. [4] give a PTAS for CVRP when Q is bounded by $O(\log n / \log \log n)$ and Adamaszek, Czumaj, and Lingas [1] improve this to a PTAS when Q is bounded by $2^{\log^{f(\varepsilon)}(n)}$ for some function $f(\varepsilon)$ depending on $\varepsilon > 0$.

Better-than-2 approximations are known for random instances of CVRP. Namely, if the depot is located at the origin $(0,0)$ and the clients are sampled uniformly at random from the square $[0,1]^2$, then algorithms with asympotically-almost-surely approximation guarantees (over the randomness in the input) better than 2 are known: first a 1.955-approximation by Bompadre, Dror, and Orlin [8], then a 1.915-approximation by Mathieu and Zhou [19], and then a 1.55-approximation by Nie and Zhou [22]. Finally, Das and Mathieu [12] give a QPTAS for CVRP, i.e., a $(1+\varepsilon)$-approximation for any constant $\varepsilon > 0$ running in quasi-polynomial-time $n^{\mathrm{poly}(\log n)}$. This is a strong evidence that CVRP might admit a PTAS.

However, despite the efforts of several researchers, the best known (polynomial-time) approximation for CVRP with no restrictions is the relatively-simple (modulo the non-trivial PTAS for Euclidean TSP) $2+\varepsilon$ approximation described before. In this paper for the first time we break this 2-approximation barrier for CVRP. Namely, we obtain the following main result.

Theorem 1.2 (Main result). *For some absolute constant $\tau > 10^{-5}$, there is a polynomial-time $(2-\tau)$-approximation for CVRP.*

Let $\mathrm{rlb} = (1-\gamma)\mathrm{opt}$ for some $\gamma \in [0,1]$. The above algorithm provides a $2+\varepsilon-\gamma$ approximation, which is good enough when γ is sufficiently large (i.e., the radial lower bound is sufficiently far from opt). We combine this with a new and drastically different algorithm which achieves a $1+\varepsilon+O(\sqrt[5]{\gamma})$ approximation, hence good when γ is sufficiently small (i.e., the radial lower bound is sufficiently close to opt). At a technical level, and interesting and novel aspect of our new algorithm is the use of matroid intersection. An overview of our approach is given in section 1.2.

1.1 Related Work. A natural generalization of CVRP, the *Unsplittable CVRP*, is obtained by letting each client $v \in \mathcal{C}$ have an integer demand $d(v) \in [0, Q]$ and requiring that for each tour T_i in a feasible solution one has $\sum_{v \in \mathcal{C}(T_i)} d(v) \leq Q$. Grandoni, Mathieu, and Zhou give the current-best $(2+\varepsilon)$-approximation [14], improving

[1]This can be easily derandomized by trying all the options.

over an earlier 2.694-approximation [13] which is for a generalization of Unsplittable CVRP, namely, the metric unsplittable CVRP (see the end of this section for more on metric unsplittable CVRP). We exploit several ideas of [14] in this work (see Section 1.2). Note that it is not possible to approximate in polynomial-time Unsplittable CVRP below a factor of 1.5 unless $\mathbf{P} = \mathbf{NP}$. Indeed, consider the case where all clients $\mathcal{C}$ are collocated at distance $1/2$ from r and the total demand is $2Q$. Then determining whether opt ≤ 2 or opt ≥ 3 would solve the Partition problem.

Capacitated Vehicle Routing has been studied in other metrics. When the capacity Q can be regarded as a fixed constant, Becker, Klein, and Schild give a PTAS in (shortest path metrics of) planar graphs [6] and Becker, Klein, and Saulpic give a PTAS in graphs with bounded highway dimension as well as exact algorithms for graphs of bounded treewidth [5]. Cohen-Addad et al. obtained efficient PTASes for constant Q in graphs of bounded treewidth, bounded highway dimension, and bounded genus plus a QPTAS for minor-free metrics which includes planar graphs [10].

For unbounded Q in restricted metrics, Jayaprakash and Salavatipour give a QPTAS for Capacitated Vehicle Routing in graphs of bounded treewidth, bounded highway dimension, and bounded doubling dimension. The latter improves on a QPTAS in doubling metrics by Khachay and Ogorodnikov when Q is polylogarithmic in the number of clients. More recently, Matheiu and Zhou give a true PTAS for unbounded Q for Capacitated Vehicle Routing instances in trees [20].

Another natural generalization of CVRP, *Metric CVRP*, is obtained by letting $c : (\mathcal{C} \cup \{r\}) \times (\mathcal{C} \cup \{r\}) \to \mathbb{R}_{\geq 0}$ be any metric. Using an ρ_{tsp}-approximation for metric TSP in combination with the approach in [15] described before gives an $\rho_{tsp} + 1$ approximation for Metric CVRP. In particular one gets a 2.5 approximation using the Metric TSP algorithm by Christofides [9] and Serdyukov [27]. Following the recent breakthroughs by Karlin, Klein, and Oveis-Gharan [17, 16], this has been improved to a constant slightly smaller than $3/2$ resulting in a guarantee slightly better than 2.5 for Metric CVRP.

Finally, in 2023 a new approach for Metric CVRP was introduced by Blauth, Traub, and Vygen [7]. In more detail, let γ be defined such that rlb $= (1 - \gamma) \cdot$ opt, where rlb is the radial lower bound defined earlier. The approximation guarantee of the algorithm in [15] is indeed at most $\rho_{tsp} + 1 - \gamma$. If this is not already substantially better than $\rho_{tsp} + 1$, then γ must be small. In this case, intuitively for most tours $T_i^* \in$ OPT, most of the clients $\mathcal{C}(T_i^*)$ are very close to the *peak* of T_i^*, i.e., the client $v \in \mathcal{C}(T_i^*)$ at maximum distance $c(r, v)$ from the depot. With such a well-structured instance, the authors are able to find cheaper TSP tours. Namely, they show how to find a TSP tour T with $c(T) \leq (1 + f(\gamma)) \cdot$ opt where $f(\gamma)$ is a function that vanishes as $\gamma \to 0$. Using T along with the tour splitting heuristic yields an approximation guarantee of $2 + f(\gamma) - \gamma$. Taking the better of the two algorithms yields an approximation with guarantee $\min\{\rho_{tsp} + 1 - \gamma, 2 + f(\gamma) - \gamma\}$. The worst case over all γ depends on ρ_{tsp} and the particular function f, but is a constant-factor smaller than $\rho_{tsp} + 1$ for any constant ρ_{tsp}. In particular, they obtain a $\left(2.5 - \frac{1}{3000}\right)$-approximation (or slightly better using the result in [17, 16]). Unfortunately, their approach does not lead to a better-than-2 approximation factor for the Euclidean case. Indeed, it is clear from the details in [7] that $f(\gamma)$ vanishes at a much slower rate than γ (i.e. $f(\gamma)$ vanishes like $O(\gamma^c)$ for some constant $c < 1$). Hence for $\rho_{tsp} = 1 + \varepsilon$, their approach does not provide better than a 2-approximation for any constant $\varepsilon > 0$.

One can also combine the above two generalizations, hence obtaining the Unsplittable Metric CVRP. As mentioned in [15], the algorithm for (Metric) CVRP can be adapted to unplittable case to give an $\rho_{tsp} + 2 \leq 3.5$ approximation (see [2] for a full argument). The improved Metric CVRP approximation in [7] gives an approximation guarantee of $3.5 - 1/3000$. Friggstad et al. [13] give a different approach to finding both a low-cost tour and a better tour-splitting approach for unsplittable CVRP to give an $\rho_{tsp} + \ln(2) + \varepsilon \approx 3.194 + \varepsilon$ approximation for any constant $\varepsilon > 0$. The latter result can be slightly improved further using the approach in [7].

We are not aware of any prior use of matroid intersection in the framework of (Metric) CVRP. However a similar matroid intersection approach was previously used to design a 2-approximation for a variant of multiple-depot TSP path [25].

1.2 Overview of Our Approach. W.l.o.g. we can assume that, for any $v \in \mathcal{C}$, $c(r, v) > 0$ (i.e., no client is collocated with r). Indeed clients collocated with r can be covered with separate tours of cost 0. By standard reductions (see Theorem 1.6), we can also assume that the distance $c(r, v)$ between the depot r and any client $v \in \mathcal{C}$ is between 1 and $D = O_\varepsilon(1)$. These assumptions introduce a $1+\varepsilon$ factor in the approximation. Furthermore, we can assume that the number Λ of tours in some optimal solution OPT is lower bounded by an arbitrarily large

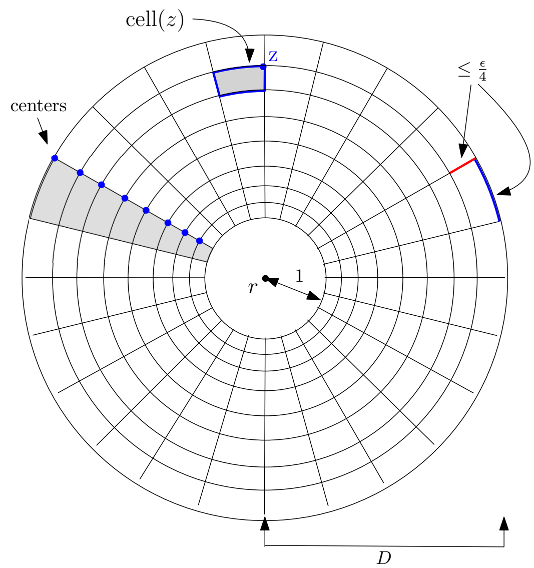

Figure 1.1: Partitioning of the region containing clients into cells. Some of the cells are shown in the gray region with their centers shown with blue points.

constant: indeed otherwise a simple modification to a PTAS for Euclidean TSP provides a PTAS for CVRP (see, e.g., [4]). Observe that the optimum solution has cost opt $\geq 2\Lambda$. Therefore in the approximation factor the cost of $O_\varepsilon(1)$ tours of length $O(D) = O_\varepsilon(1)$ can be neglected. Finally, we can assume that Q is lower bounded by a sufficiently large constant, otherwise a PTAS exists [4].

For simplicity, let us next neglect the $1+\varepsilon$ factors due to the above reductions and to the PTAS for Euclidean TSP. Observe that, if rlb $\leq (1-\gamma)\cdot$ opt for some constant $\gamma > 0$, then the existing approaches already give a tsp + rlb $\leq (2-\gamma)\cdot$ opt approximation.

So we can assume w.l.o.g. that this is not the case, and, for the sake of this overview, imagine $\gamma > 0$ as being a very small constant. We next illustrate how to get an approximation factor very close to 1 for this setting. The τ from Theorem 1.2 simply comes from a more careful balancing of the different constants in the following discussion.

Let us partition the ball $B(r, D)$ of radius D center at r (recall that such ball contains all the clients $\mathcal{C}$) into *cells* similarly to [14]: intuitively, we draw rays centered at r where consecutive rays induce a sufficiently small angle $\theta = \Theta(\frac{\varepsilon}{D})$, and we draw circumferences centered at r with radius between 1 and D such that consecutive circumferences differ in radius by $r' = \Theta(\varepsilon)$ (see also Figure 1.1). Here $\varepsilon \ll \gamma$. This induces $K = O(\frac{D^2}{\varepsilon^2}) = O_\varepsilon(1)$ regions that we call *cells*. Notice that each client belongs to precisely one cell (breaking ties arbitrarily, but favoring cells which are closer to r). Furthermore, by choosing θ and r' appropriately, we can enforce that the perimeter of each cell and the maximum distance among any two points in the cell is at most ε. We choose some point z (the precisely location is not critical) of each cell as the *center* of this cell. Let cell(z) be the cell corresponding to center z, $\mathcal{C}(z)$ be the clients in cell(z), and Z be the set of centers. In the following we will consider the expansion of the associated graph G where we add the nodes Z with the respective edges and costs.

For an optimal tour $T^* \in$ OPT, let peak(T^*) be the client in $\mathcal{C}(T^*)$ which is furthest away from r (breaking ties arbitrarily), and let $z(T^*)$ be the center of the cell containing peak(T^*) (*peak center* of T^*). Let $\delta, \delta' > 0$ be two small enough constants. For the sake of this overview, we set $\delta = \Theta(\sqrt[5]{\gamma})$ and $\delta' = \Theta(\sqrt[5]{\gamma^2})$ so that $\delta \cdot (\delta')^2 > \gamma$. We define the δ-neighboring cells of center $z \in Z$ as the cells cell(z') containing at least one point at distance at most $\delta \cdot c(r, z) + \varepsilon$ from z and such that $c(r, z') \leq c(r, z)$ (see also Figure 1.4). The *peak clients* $\mathcal{C}_{peak}(T^*)$ of T^* are the clients in $\mathcal{C}(T^*)$ contained in a δ-neighboring cell of $z(T^*)$. The remaining clients $\mathcal{C}_{lfto}(T^*) := \mathcal{C}(T^*) \setminus \mathcal{C}_{peak}(T^*)$ are the *leftover* clients of T^*. We also let $\mathcal{C}_{peak}$ (resp, $\mathcal{C}_{lfto}$) be the set of all the peak (resp., leftover) clients. In the following, for a subgraph S (usually a path or a tour), $\mathcal{C}_{peak}(S) = \mathcal{C}(S) \cap \mathcal{C}_{peak}$ and $\mathcal{C}_{lfto}(S) = \mathcal{C}(S) \cap \mathcal{C}_{lfto}$. The assumption that rlb $\geq (1-\gamma)\cdot$ opt imposes the following helpful property on OPT:

(a) One has $\text{rlb}(\mathcal{C}_{peak}) \geq (1-\Theta(\delta)) \cdot \text{rlb}$ and $\text{rlb}(\mathcal{C}_{lfto}) \leq O(\delta') \cdot \text{opt}$, where $\text{rlb}(W) := \frac{1}{Q}\sum_{v\in W} 2\cdot c(r,v)$ is the portion of the radial lower bound corresponding to $W \subseteq \mathcal{C}$.

To see why the above property holds, suppose by contradiction that the *bad* tours T^* where at least $\delta' \cdot Q$ clients are not peak clients, have total cost at least $\delta' \cdot \text{opt}$ (the other tours are called *good*). For each such bad T^*,

$$\text{rlb}(\mathcal{C}(T^*)) \leq ((1-\delta') + \delta' \cdot (1-\delta)) \cdot 2 \cdot c(r, z(T^*)) \leq (1 - \delta \cdot \delta') \cdot c(T^*),$$

where the first inequality follows from the fact that at least $\delta' \cdot Q$ clients on T^* have distance at most $(1-\delta)\cdot c(r, z(T^*))$ from r where each such clients contribute much less than $c(T^*)$ to rlb (see Claim 2.6 for a proof), and the rest (at most $(1-\delta')\cdot Q$ many) of the clients have distance at most $c(r, z(T^*))$ from r. As a consequence one gets $\text{rlb} \leq \text{opt} - \delta' \cdot \text{opt} + (1-\delta\cdot\delta')\cdot\delta'\cdot\text{opt} < (1-\gamma)\cdot\text{opt}$, a contradiction. One consequence is that

$$\begin{aligned}
\text{rlb}(\mathcal{C}_{peak}) &\geq (1-\delta)(1-\delta') \sum_{T^*\in\text{OPT}, T^*\text{good}} 2\cdot c(r, z(T^*)) \\
&\geq (1-\delta)(1-\delta') \sum_{T^*\in\text{OPT}, T^*\text{good}} \text{rlb}(\mathcal{C}(T^*)) \\
&\geq (1-\delta)(1-\delta')^2 \cdot (\text{rlb} - \delta'\text{opt}) = (1-\Theta(\delta))\cdot\text{rlb}.
\end{aligned}$$

Another consequence is that

$$\text{rlb}(\mathcal{C}_{lfto}) \leq \sum_{T^*\in\text{OPT}, T^* \text{ bad}} \text{rlb}(V(T^*)) + \delta' \sum_{T^*\in\text{OPT}, T^* \text{ good}} c(T^*) \leq O(\delta')\text{opt}.$$

In order to illustrate our main ideas, let us make for a moment the following second assumption (we will later discuss how to get rid of it):

(b) For each tour $T^* \in \text{OPT}$, its peak clients $\mathcal{C}_{peak}(T^*)$ are collocated in its peak center $z(T^*)$.

We next sketch how to get a $1 + O(\sqrt[5]{\gamma})$ approximation based on (a) and (b). We remark that, if all the clients are located at the centers, then a PTAS is easy to derive (see, e.g., [14]). However this is not known in the above setting due to the leftover clients.

Exploiting the fact that there are only $O_\varepsilon(1)$ cells, we can guess the number t_z of optimal tours that have each center z as its peak center. We can also guess how many clients n^z located at z are peak clients. Even more, since the clients located at z are indistinguishable, we can arbitrarily choose any n^z such clients to be peak clients and let the remaining ones be leftover clients (together with the clients in the same cell which are not located in z). Let $\mathcal{C}_{peak}(z)$ be the peak clients located at z: observe that obviously $n^z = |\mathcal{C}_{peak}(z)| \leq t_z \cdot Q$; we will use this observation later. Notice that so far we used Property (b) only.

At this point, our first main novel idea comes into play. We can interpret each optimal tour T^* as two paths between r and $z(T^*)$: each such path collects some leftover clients, and furthermore the tour collects some peak clients in $z(T^*)$; clearly the total number of collected clients is at most Q. So our goal is to compute a collection of such paths that visit all the leftover clients and a partition of the peak clients in each center into subsets, such that it is possible to subdivide all the paths and subsets into triples containing 2 paths and one subset of peak clients that altogether involve at most Q clients. At the same time, the total length of the paths should be minimized.

We relax the above problem as follows: we compute a forest of trees of minimum total cost (in terms of the total length of their edges) such that each tree contains r and some center z, the number of trees containing z is precisely $2 \cdot t_z$, and the trees altogether span all the leftover clients. This problem can be reduced to matroid intersection, hence it is solvable optimally in polynomial time (see, e.g., Theorem 41.6 in [26]). In more detail, we replace the peak nodes located at z with $2 \cdot t_z$ nodes located there (we call these nodes the copies of z), and we replace r with $2 \cdot t_z$ copies of r for each center z. Then one matroid is a graphical matroid plus the extra constraint that no tree contains more than one copy of a center, see [23] for a proof this is a matroid. Similarly, the other matroid is a graphical matroid plus the extra constraint that each tree contains at most one copy of r. This way, a base that is common to both matroids is a forest where each tree has exactly one copy of r and exactly one copy of some center z.

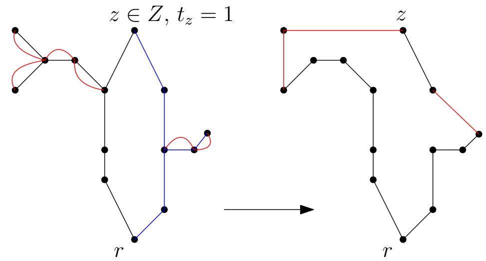

Figure 1.2: Left picture shows two trees (one in black edges and one in blue edges) both containing r and a center $z \in Z$. In each tree, we double the edges not on the unique r, z-path, shown in red edges. Right picture shows two edge-disjoint r, z-path obtained from the trees and the added edges and shortcutting whenever necessary.

Let $F_1, \ldots, F_q$ be the computed trees. We remark that $\sum_{i=1}^{q} c(F_i) \leq \mathrm{opt}$ since OPT induces a feasible base for the considered matroid intersection. Next we exploit Property (a) to transform each such tree with center z into an r-z path with a negligible increase of the overall cost. Consider any tree F_i in such forest containing a center z_i. Remember that F_i also contains r, hence in particular it contains one r-z_i path $\tilde{P}_i$. We duplicate all the edge of F_i *not* in $\tilde{P}_i$ and via shortcutting we transform the resulting multi-graph into an r-z_i path P_i without any further increase of the cost (see also Figure 1.2).

Interpreting both F_i and $\tilde{P}_i$ as sets of edges, one has $c(P_i) \leq c(F_i) + c(F_i \setminus \tilde{P}_i)$ and trivially $c(\tilde{P}_i) \geq c(r, z_i)$. By Properties (a) and (b),

$$\sum_{T^* \in \mathrm{OPT}} 2 \cdot c(r, z(T^*)) \overset{(b)}{\geq} \sum_{T^* \in \mathrm{OPT}} \mathrm{rlb}(\mathcal{C}_{peak}(T^*)) \overset{(a)}{\geq} (1 - \Theta(\delta)) \cdot \mathrm{rlb} = (1 - \Theta(\delta)) \cdot \mathrm{opt}.$$

Then,

$$\sum_{i=1}^{q} c(F_i \setminus \tilde{P}_i) \leq \sum_{i=1}^{q} (c(F_i) - c(r, z_i)) = \sum_{i=1}^{q} c(F_i) - \sum_{T^* \in \mathrm{OPT}} 2 \cdot c(r, z(T^*)) \leq \Theta(\delta)\mathrm{opt} = O(\sqrt[5]{\gamma})\mathrm{opt},$$

where in the last inequality we used the fact that $\sum_i c(F_i) \leq \mathrm{opt}$.

Consider now the obtained collection $P_1, \ldots, P_q$ of paths, where each P_i is an r-z_i path for some center z_i. Next, we need to turn these paths into a feasible CVRP solution. Notice that, unlike similar steps in prior work [15, 7], we cannot afford to incur a cost of rlb here since this would not be sufficient to obtain a better than 2 approximation. Recall that these paths together span all the leftover clients. We remark that each P_i might contain more than Q clients, which makes complicated to combine such paths together into feasible tours. Indeed, for technical reasons, it would be convenient if each such path contained at most $\rho \cdot Q$ (that we assume to be integer for simplicity) leftover clients for some small enough constant $0 < \rho < \frac{1}{2}$, that for the sake of this presentation we fix to $\sqrt[5]{\gamma}$ in this overview. We can enforce this property with a negligible increase of the cost by exploiting Property (a) in combination with the tour splitting heuristic (see also Figure 1.3 and Lemma 1.8). In more detail, consider any such P_i with more than $\rho \cdot Q$ clients and let $r, v_1, \ldots, v_\ell, z_i$ be the ordered sequence of nodes along P_i. We choose a random[2] offset $s \in \{1, \ldots, \rho \cdot Q\}$ and split P_i into subpaths $P_i^0 = r, v_1, \ldots, v_s$, $P_i^1 = v_{s+1}, \ldots, v_{s+\rho Q}$, $P_i^2 = v_{s+\rho \cdot Q+1}, \ldots, v_{s+2 \cdot \rho \cdot Q}$, …, $P_i^{h_i+1} = v_{s+h_i \cdot \rho Q+1}, \ldots, v_\ell, z_i$ (each one containing at most $\rho \cdot Q$ leftover clients). Then for each such subpath P_i^j excluding the last one we add the edge $\{r, v_{s+j \cdot \rho \cdot Q}\}$ between the depot and its last node. Furthermore for each such path P_i^j excluding the first one, we add the edge $\{r, v_{s+(j-1) \cdot \rho \cdot Q+1}\}$ between the depot and its first node. Thanks to Property (a), the cost of the added edges is negligible. More precisely, the expected increase of the cost w.r.t. $c(P_i)$ can be upper bounded by $\frac{1}{\rho Q} \sum_{v \in \mathcal{C}(P_i)} 2 \cdot c(r, v) = \frac{1}{\rho} \cdot \mathrm{rlb}(\mathcal{C}(P_i))$, i.e., $\frac{1}{\rho}$ times the radial lower bound associated with $\mathcal{C}(P_i) = \mathcal{C}_{lfto}(P_i)$. By Property (a) one has that

$$\sum_{v \in \mathcal{C}_{lfto}} \mathrm{rlb}(\{v\}) = \sum_{i=1}^{q} \sum_{v \in \mathcal{C}_{lfto}(P_i)} \mathrm{rlb}(\{v\}) \overset{(a)}{\leq} \Theta(\delta') \cdot \mathrm{opt}.$$

[2] Also in this case the derandomization is trivial.

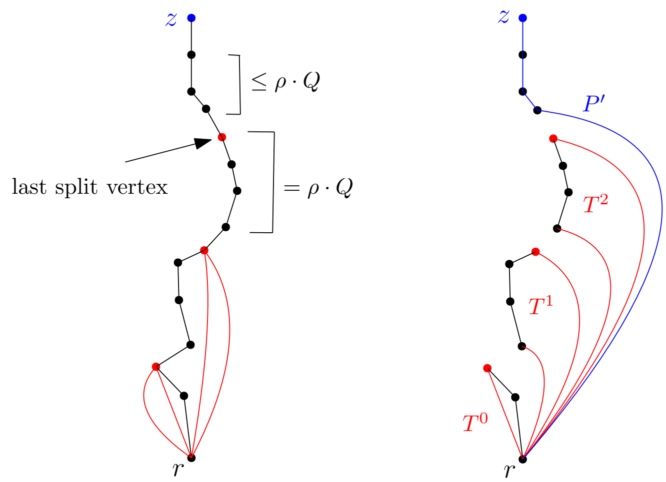

Figure 1.3: On the left, an r-z path P. On the right, the resulting tours T^0, T^1, T^2 and one r-z path P', each one containing at most $\rho \cdot Q$ clients for a parameter $0 < \rho < 1$.

Hence the total increase of the cost is at most $\frac{O(\delta')}{\rho} \cdot \mathrm{opt} \leq O(\sqrt[5]{\gamma}) \cdot \mathrm{opt}$. The added edges transform each subpath P_i^j excluding the last one into a tour T_i^j containing at most $\rho \cdot Q < Q$ clients in total: these tours are simply added to the final solution. The remaining $P_i' := P_i^{h_i+1} \cup \{r, v_{s+h_i \cdot \rho Q}\}$ is an r-z_i path containing at most $\rho \cdot Q$ clients as desired.

It remains to handle the leftover clients contained in the subpaths P_i' plus all the peak clients. This is relatively easy to do. Let us focus on a specific center z. Recall that there are $2 \cdot t_z$ paths of type P_i' incident to z. We arbitrarily group such paths in pairs, and for each such pair $(P^{(a)}, P^{(b)})$ we add to the solution the tour $P^{(a)} \cup P^{(b)}$. The obtained tour covers the (at most $2\rho Q$ many) leftover clients $\mathcal{C}_{lfto}(P^{(a)}) \cup \mathcal{C}_{lfto}(P^{(b)})$ plus a number of distinct peak clients in z to reach Q clients in total (assuming that there are enough peak clients left, otherwise we simply add all the remaining peak clients at z). Notice that all the leftover clients are already covered at this point. If we happen to cover all the peak clients at z this way, we are done w.r.t. z. Otherwise we construct a minimal number of tours of length $2 \cdot c(r, z)$ to cover the remaining peak clients at z.

We next argue that the latter tours are cheap enough. Recall that OPT has t_z tours with peak center z, hence the peak clients at z are at most $t_z \cdot Q$. Each pair of paths $(P^{(a)}, P^{(b)})$ ending in z contain at most $2 \cdot \rho \cdot Q$ clients in total, hence the corresponding tour consumes at least $(1 - 2\rho) \cdot Q$ peak clients at z (since we did not exhaust all the peak clients at z). Since there are t_z many such tours and the total number of peak clients at z is at most $t_z \cdot Q$, the number of remaining peak clients to cover is at most $2 \cdot \rho \cdot t_z \cdot Q$. Hence we need at most $\lceil 2 \cdot \rho \cdot t_z \rceil \leq 2 \cdot \rho \cdot t_z + 1$ extra tours of cost $2 \cdot c(r, z)$ to serve them, paying in total at most $(2 \cdot \rho \cdot t_z + 1) \cdot 2 \cdot c(r, z)$. Notice that OPT must pay at least $t_z \cdot 2 \cdot c(r, z)$ for the tours with peak center at z. This means that we pay at most $2 \cdot \rho \cdot \mathrm{opt} = O(\sqrt[5]{\gamma}) \cdot \mathrm{opt}$ for the final tours, excluding at most one tour of cost $2 \cdot c(r, z)$ per center z. The cost of the latter tours is at most $O(\varepsilon) \cdot \mathrm{opt}$. Indeed each such tour costs at most $2 \cdot D$, while the optimum solution costs at least $2 \cdot \Lambda$ (recall that OPT contains Λ tours). Hence it is sufficient to assume that $\Lambda \geq \frac{1}{\varepsilon} \cdot D \cdot K = O_\varepsilon(1)$, where K is the number of cells. Altogether this gives a $1 + O(\sqrt[5]{\gamma})$ approximation as promised.

We next discuss how to get rid of Property (b). This step is more technical in spirit. We construct, for each center z, a tour $T_{\text{peak}}(z)$ containing the desired (guessed) number $n^z = |\mathcal{C}_{peak}(z)|$ of clients for z (but not the depot!) as follows. Recall $\mathcal{C}_{peak}(z)$ is the set of clients in the δ-neighboring cells of cell(z) such that each such client belongs to an optimal tour T^* where $z(T^*) = z$. For each δ-neighboring cell cell(z') of z (including cell(z) itself), we guess the number $n^z_{z'}$ of peak clients $\mathcal{C}_{peak}(z)$ which are contained in cell(z'). We set $n^z_{z'} = 0$ for the remaining cells cell(z'). Clearly $\sum_{z' \in Z} n^z_{z'} = n^z$, and this guessing involves a polynomial number of options. Now for each cell cell(z'), we compute the (roughly) cheapest tour $T_{\text{k-tsp}}(z')$ that covers any $k_{z'} := \sum_{z \in Z} n^z_{z'}$ clients inside cell(z') using a known PTAS for Euclidean k-TSP[3] [3, 29]. We remark that $T_{\text{k-tsp}}(z')$ contains a subset of the clients $\mathcal{C}(z')$ inside cell(z'). In particular, it must be $k_{z'} \leq |\mathcal{C}(z')|$. We build $T_{\text{peak}}(z)$ by cutting from each

[3]Recall that k-TSP is a variant of TSP where one searches for the shortest tour that visits k nodes (instead of all nodes).

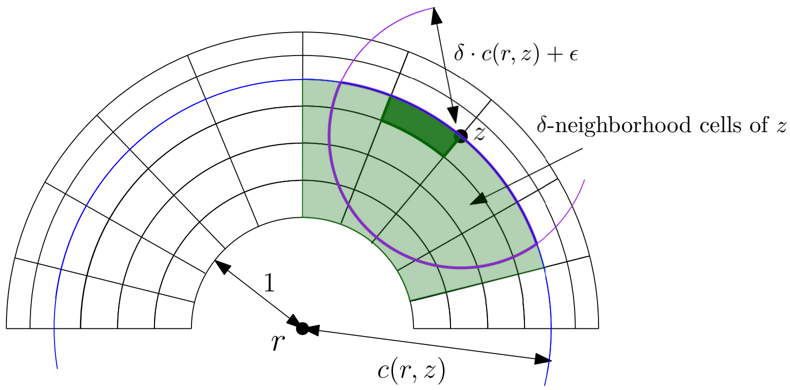

Figure 1.4: The cell cell(z) with center z is shown with the bold green region. The δ-neighborhood of z is shown with bold purple boundaries. The δ-neighboring cells of z are illustrated with light green color.

$T_{\text{k-tsp}}(z')$ a subpath containing $n^z_{z'}$ distinct clients (*fragment*), and then gluing these fragments together (see also Figure 1.5.a). Notice that the clients in the tours $T_{\text{peak}}(z)$ are not necessarily the peak clients $\mathcal{C}_{peak}$ in OPT. However for the rest of the proof we can simply consider them as peak clients, and call leftover the remaining ones.

Let us argue why the total cost of the tours $T_{\text{peak}}(z)$ is small enough. Consider any $T^* \in \text{OPT}$, and interpret it as a polygonal chain in the plane. Consider the projection[4] T^*_{peak} of T^* into the neighboring cells of $z(T^*)$, and let $c(T^*_{\text{peak}})$ be the total length of this projection. The fact that rlb $\geq (1-\gamma) \cdot$ opt imposes that $\sum_{T^* \in \text{OPT}} c(T^*_{\text{peak}}) \leq O(\sqrt[5]{\gamma}) \cdot \text{opt}$. The intuition why this holds is that, rlb $\geq (1-\gamma) \cdot$ opt implies the optimal tours with peak center z have length on average at most $(1 + O(\gamma)) \cdot 2 \cdot c(r, z)$, i.e., the optimal tours on average are almost two straight lines between r and z. Traversing these lines starting from r, the clients on T^*_{peak} are, by definition of T^*_{peak}, roughly on the last $\delta \cdot c(r, z(T^*))$ portion of these lines which implies $c(T^*_{\text{peak}}) \leq O(\delta) \cdot c(T^*)$, this is proved formally in Claim 4.5. Next consider the projections of all the T^*_{peak} inside a given cell cell(z'). We can glue these projections in a single tour by paying $3/2$ times the perimeter ε of cell(z') using the approach in [18]. Hence the gluing cost is at most $O(\varepsilon) \cdot K$ in total, where $K = O_\varepsilon(1)$ is the total number of cells. The latter gluing cost is negligible since opt $\geq 2\Lambda$, and we can assume w.l.o.g. that $\Lambda \geq K$ (otherwise a PTAS exists). Summarizing, the tours $T_{\text{k-tsp}}(z')$ will cost in total at most $(O(\sqrt[5]{\gamma}) + \varepsilon) \cdot \text{opt} = O(\sqrt[5]{\gamma}) \cdot \text{opt}$. Via a similar argument, we can glue together all the fragments of each $T_{\text{k-tsp}}(z')$ which are needed to build $T_{\text{peak}}(z)$ at a negligible cost. Indeed, we need to glue at most K fragments per center z, using edges of length at most $O(D)$. So the total cost is at most $O(D \cdot K^2) = O(\varepsilon) \cdot \text{opt}$ by assuming that Λ is large enough, this is proved formally in Lemma 4.6.

It remains to describe how to use these tours $T_{\text{peak}}(z)$, containing n^z clients each and of total cost $O(\sqrt[5]{\gamma}) \cdot \text{opt}$, in place of Property (b). We can compute the forest $F_1, \ldots, F_q$ as in the previous description. The total cost of the forest is now $(1 + O(\varepsilon)) \cdot \text{opt}$, where the extra $O(\varepsilon)$ factor comes from the fact that we insist on each tree F_i in the forest containing the corresponding peak center z_i (while z_i might not belong to OPT). Then we can split each F_i into paths $P_i^0, \ldots, P_i^{h_i+1}$, derive the tours $T_i^0, \ldots, T_i^{h_i}$ and the path P'_i as before. Recall that in the above description we considered t_z pairs of r-z paths $(P^{(a)}, P^{(b)})$ collecting $|\mathcal{C}_{lfto}(P^{(a)})| + |\mathcal{C}_{lfto}(P^{(b)})| \leq 2 \cdot \rho \cdot Q$ leftover clients in total. Hence we need to collect (up to) $Q - |\mathcal{C}_{lfto}(P^{(a)})| + |\mathcal{C}_{lfto}(P^{(b)})|$ extra peak clients corresponding to z: we do so by cutting a distinct fragment of $T_{\text{peak}}(z)$ containing the desired number of clients, and then connecting the endpoints of this fragment with z at cost $O(\delta) \cdot c(r, z) = O(\sqrt[5]{\gamma}) \cdot c(r, z)$ (see also Figure 1.5.b). Hence the latter connection costs $O(\sqrt[5]{\gamma}) \cdot \text{opt}$ in total. The same approach also works to cover the final (up to) $2 \cdot \rho \cdot t_z \cdot Q$ peak clients at z as in the previous discussion, at a total cost of $O(\delta) \cdot \text{opt} = O(\sqrt[5]{\gamma})\text{opt}$. Altogether, we obtain a $1 + O(\sqrt[5]{\gamma})$ approximation as promised.

Balancing the latter factor with the $2-\gamma$ factor obtained trivially in the complementary case rlb $\leq (1-\gamma) \cdot \text{opt}$, one obtains a strictly better than 2 approximation.

[4]By the projection of a tour T^* into a cell cell(z), we mean the restriction of T^* inside cell(z). In particular, the projection consists of a collection of polygonal chains that lie inside cell(z) and whose endpoints lie on the boundary of cell(z).

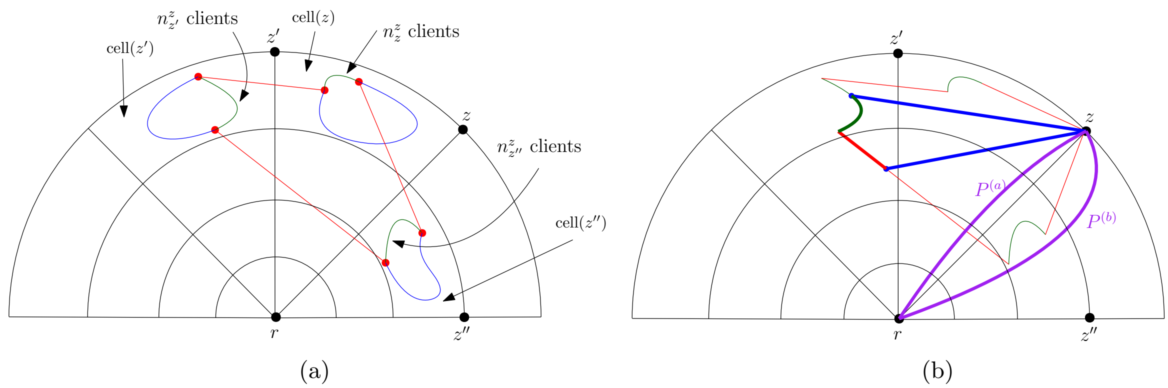

Figure 1.5: (a) Green and blue edges define the tours $T_{\text{k-tsp}}(z')$ while red and green edges define the tour $T_{\text{peak}}(z)$. (b) We connect the endpoints of a fragment of $T_{\text{peak}}(z)$ (shown in bold red and green) to z using the bold blue edges to form a tour including z. Then, paths $P^{(a)}$ and $P^{(b)}$ (shown in bold purple) plus this tour form a tour in our final solution.

1.3 Preliminaries. Throughout the paper, the union of any two sets of edges is preformed with multiplicities, i.e., if $e \in F \cap F'$ then we have two copies of e in $F \cup F'$. For a subgraph H, we denote by $V(H)$ and $E(H)$ the vertices and edges of H, respectively. We define $\mathcal{C}(H)$ to be the set of clients in $V(H)$. We allow multiple copies of an edge in a subgraph so throughout the paper by subgraph we mean multi-subgraph. We define the cost of subset of edges E' to be the sum of the cost of edges in E' and denote it by $c(E') := \sum\limits_{e \in E'} c(e)$. By a *tour* we mean a cycle that contains some vertices of V. Sometime we define a path or a tour F on V with the help of some points in $\mathbb{R}^2$ that does not belong to V. We let $\mathcal{C}(F) := V(F) \cap \mathcal{C}$. For a tour T, we define $\text{peak}(T)$ to be the client in $\mathcal{C}(T)$ with the maximum distance from r (breaking ties arbitrarily).

Let $P = v_1, v_2, \ldots, v_k$ be a path where $v_i \in \mathcal{C}$. We can view P in the plane as well. Note P is a collection of straight lines between v_i and v_{i+1} for $1 \le i \le k-1$. Let $x \in \mathbb{R}^2$ be a point on the straight line between v_i and v_{i+1} for some $i = 1, \ldots, k-1$. Then, we define the cost of subpath of P between v_j and x for any $j \le i$ to be the cost of the subpath $v_j, v_{j+1}, \ldots, v_i$ plus $c(v_i, x)$. We denote this subpath by $P_{v_j,x}$. Similarly, we define the cost of subpath of P between x and v_j for any $j > i$. Since P is a set of (edges) straight lines and c is the Euclidean metric, we have $c(P) = c(P_{v_1,x}) + c(P_{x,v_k})$.

Given an ordered sequence of clients $v_1, v_2, \ldots, v_k$, a *fragment* of size $1 \le \ell \le k$ is a contiguous subsequence, i.e., a subsequence made up of ℓ consecutive elements of $v_1, \ldots, v_k$. Furthermore, the endpoints of a fragment are the clients in the fragment with the smallest and the largest indices.

We already mentioned the radial lower bound $\text{rlb} := \frac{1}{Q} \sum\limits_{v \in \mathcal{C}} 2 \cdot c(r, v) \le \text{opt}$ and the TSP lower bound $\text{tsp} \le \text{opt}$, where tsp is the length of a shortest tour spanning $\mathcal{C} \cup \{r\}$. Recall that, for $X \subseteq \mathcal{C}$, we use $\text{rlb}(X) := \frac{1}{Q} \sum_{v \in X} 2 \cdot c(r, v)$ to denote the part of rlb restricted to X. The following lemma summarizes the discussion about the classical $2 + \varepsilon$ approximation for CVRP.

Lemma 1.3 ([15, 3, 21]). *In polynomial time one can compute a solution to a CVRP instance of cost at most* $(1 + \varepsilon) \cdot \text{tsp} + \text{rlb}$.

In the following we will need also the following other lower bound. For a tour $T^* \in \text{OPT}$, let $\text{peak}(T^*)$ be the client $v \in \mathcal{C}(T^*)$ (the *peak* client or simply *peak* of T^*) at maximum distance from r (breaking ties arbitrarily). Then the *peak lower bound* is

$$\text{plb} := \sum_{T^* \in \text{OPT}} 2 \cdot c(r, \text{peak}(T^*)).$$

Notice that $\text{plb} \le \text{opt}$ since each tour $T^* \in \text{OPT}$ can be split into two paths between r and $\text{peak}(T^*)$, each one of length at least $c(r, peak(T^*))$. Where remark that plb is stronger lower bound than rlb, indeed

$$\text{rlb} \le \sum_{T^* \in \text{OPT}} \frac{1}{|\mathcal{C}(T^*)|} \cdot \sum_{v \in \mathcal{C}(T^*)} 2 \cdot c(r, v) \le \sum_{T^* \in \text{OPT}} 2 \cdot c(r, \text{peak}(T^*)) = \text{plb}.$$

The following lemma summarizes the above discussion.

LEMMA 1.4 (Lower bounds). *One has* rlb $\leq$ plb $\leq$ opt *and* tsp $\leq$ opt.

We define **regret distances** on $(\mathcal{C} \cup \{r\}) \times \mathcal{C}$ which will be important for us later. For two vertices u and v, we define its regret to be $\mathrm{regret}(u, v) := c(r, u) + c(u, v) - c(r, v)$. It is easy to verify that $\mathrm{regret}(.,.)$ satisfies non-negativity and triangle inequality but note we may have $\mathrm{regret}(u, v) \neq \mathrm{regret}(v, u)$. For a path $P = v_1, v_2, \ldots, v_m$, the regret is defined as the sum of regret of edges of P, i.e., $\mathrm{regret}(P) := \sum_{i=1}^{m-1} \mathrm{regret}(v_i, v_{i+1})$. We have the following easy to verify fact about the regret of paths.

FACT 1.5. *Consider a path $P = r, v_1, v_2, \ldots, v_m$ starting at r. Then, we have* $\mathrm{regret}(P) = c(P) - c(r, v_m)$.

We view a tour $T^* \in \mathrm{OPT}$ as two paths P_{T^*}, P'_{T^*} starting from r and ending at the peak $\mathrm{peak}(T^*)$. We define the regret of OPT as

$$
\begin{aligned}
\mathrm{regret}(\mathrm{OPT}) &:= \sum_{T^* \in \mathrm{OPT}} \mathrm{regret}(P_{T^*}) + \mathrm{regret}(P'_{T^*}) \\
&= \sum_{T^* \in \mathrm{OPT}} c(P_{T^*}) + c(P'_{T^*}) - 2 \cdot c(r, \mathrm{peak}(T^*)) \qquad (1.1) \\
&= \sum_{T^* \in \mathrm{OPT}} c(T^*) - 2 \cdot c(r, \mathrm{peak}(T^*)) = \mathrm{opt} - \mathrm{plb}.
\end{aligned}
$$

It is crucial for the analysis of our algorithm that the CVRP instance has a bounded aspect ratio. We will use the following result due to [1] (see also [14]).

THEOREM 1.6 (Reduction to bounded aspect ratio). *For any constant $\varepsilon > 0$, given a polynomial-time α approximation for CVRP under the assumption that $1 \leq c(r, v) \leq D = (\frac{1}{\varepsilon})^{\frac{1}{\varepsilon}}$ for each $v \in \mathcal{C}$, there is a polynomial-time $\alpha(1 + \varepsilon)$-approximation for CVRP with no restriction.*

We next assume that the input instance has bounded aspect radius as in Theorem 1.6. In particular all the clients are contained in an annulus centers at r with inner radius 1 and outer radius D. We call this instance, a *D-bounded instance* of CVRP. We partition this annulus into regions called cells similarly to [14] (see also Figure 1.1): we call this a cell splitting. In more detail, we consider circumferences with center r where the i-th circumference has radius $1 + \frac{\varepsilon}{4} \cdot i$ for integers i in $0 \leq i \leq K_1 := \frac{4 \cdot D}{\varepsilon}$. Next, we draw rays from r with the angle between any two consecutive rays equal to $\frac{2 \cdot \pi}{K_2}$ where $K_2 := \lceil \frac{8 \cdot \pi \cdot D}{\varepsilon} \rceil$, i.e., these rays partition any circle with center r into K_2 sectors.

The intersection of the region between any two consecutive circumferences and the region between any two consecutive rays, is called a *cell*. We call any point on the plane that is in the intersection of a circumference (except the circumference with radius 1) and a ray a *center*. Denote by Z the set of all centers. Note that for each cell A, there are up to 4 centers that are on the boundary of A. Among these four centers, we assign the one whose polar coordinate has the largest distance from the origin (i.e., r) and the smallest angle as the *center of A*. For $z \in Z$, we define $\mathrm{cell}(z)$ to be the cell whose center is z. Each client belongs to precisely one cell, breaking ties arbitrarily, but favoring cells whose center is closer to r. We denote by $\mathcal{C}(z)$ the set of clients in $\mathrm{cell}(z)$. The length of the boundary of $\mathrm{cell}(z)$ is denoted by $\mathrm{bd}(z)$. See Figure 1.1 for an illustration. For a tour T, we define the *peak center* of T to be the center of the cell containing $\mathrm{peak}(T)$. For the rest of the paper, we will consider the expansion of the associated graph where we add the nodes Z with the respective edges and costs.

FACT 1.7. *The following properties hold:*

1. *The maximum distance between any two points in* $\mathrm{cell}(z)$ *is at most* $\frac{\varepsilon}{2}$ *and* $\mathrm{bd}(z) \leq \varepsilon$.
2. *For any $x \in \mathbb{R}^2$ in* $\mathrm{cell}(z)$*, we have $c(r, x) \leq c(r, z)$.*
3. *The number of centers/cells is at most* $K := K_1 \cdot K_2 = \frac{4 \cdot D}{\varepsilon} \cdot \lceil \frac{8 \cdot \pi \cdot D}{\varepsilon} \rceil$.

Proof. Properties (2) and (3) are trivial.

Proof of property (1): Consider $\mathrm{cell}(z)$ where $c(r, z) = 1 + \frac{\varepsilon}{4} \cdot j$ for some $1 \leq j \leq K_1$. Then, the boundary of $\mathrm{cell}(z)$ on the j-circumference has length $c(r, z) \cdot \frac{2 \cdot \pi}{K_2} \leq \frac{\varepsilon}{4}$. Similarly, the length of the boundary of $\mathrm{cell}(z)$

on the $(j-1)$-th circumference is at most $\frac{\varepsilon}{4}$. The length of the other two boundaries of cell(z) is exactly $\frac{\varepsilon}{4}$ by definition of our circumferences. This proves bd$(z) \le \varepsilon$ and the distance between any two points in cell(z) is at most $\frac{\varepsilon}{4} + \frac{\varepsilon}{4} = \frac{\varepsilon}{2}$. ∎

As discussed in Section 1.2, given a path we use a standard randomized procedure to divide the path into fragments of a particular size. A similar procedure was used in [13]. For an illustration see Figure 1.3.

LEMMA 1.8 (Path partitioning). *Given a path $P = r, v_1, v_2 \dots, v_k, z$ where $v_i \in \mathcal{C}$ for $i = 1, \dots, k$, $z \in Z$ (a center), and an integer fragment size $\rho \cdot Q$ for some $0 < \rho < 1$, there is a polynomial-time algorithm that turns P into some tours $T^0, \dots, T^h$ each one containing r and at most $\rho \cdot Q$ clients, and exactly one r-z path P' containing at most $\rho \cdot Q$ clients, such that the tours and P' span all the clients of P, i.e., $\mathcal{C}(P) = \mathcal{C}(T^0) \cup \ldots \cup \mathcal{C}(T^h) \cup \mathcal{C}(P')$. Furthermore, the total cost of these tours and P' is at most $c(P) + \frac{1}{\rho \cdot Q} \cdot \sum_{i=1}^{k} 2 \cdot c(r, v_i)$.*

Proof. First, we give a randomized algorithm and then we discuss how to derandomize it.

If $k \le \rho \cdot Q$, return $P' := P$. Otherwise, choose a random offset $s \in \{1, 2, \dots, \rho \cdot Q\}$. We call v_i *a split client* if and only if there is an integer $\ell \ge 0$ such that

$$i = s + \ell \cdot \rho \cdot Q$$

The probability, over the random choices of s, that v_i is a split client is exactly $\frac{1}{\rho \cdot Q}$. Let $v_{i_1}, v_{i_2}, \dots, v_{i_{h+1}}$ be the split clients. For every split client v_{i_j}, add two copies of $\{r, v_{i_j}\}$. Define the following tours with the help of the added edges: $\tilde{T}^0 := r, v_1, \dots, v_{i_1}, r$, $\tilde{T}^1 := r, v_{i_1}, \dots, v_{i_2}, r, \dots$, $\tilde{T}^h := r, v_{i_h}, \dots, v_{i_{h+1}}, r$. For $j = 1, \dots, h$, obtain T^j from $\tilde{T}^j$ by shortcutting v_{i_j}. We also let $T^0 := \tilde{T}^0$. Note that $c(T^j) \le c(\tilde{T}^j)$ for $j = 0, \dots, h$.

Lastly, we define the path $P'' := r, v_{i_h}, \dots, v_k, z$. By shortcutting v_{i_h}, we obtain $P' := r, v_{i_h+1}, \dots, v_k, z$ of cost no more than $c(P'')$. It is easy to see this collection of tours and P' satisfy the requirement of the lemma.

Note the total cost of the tours and P' is at most $c(P)$ plus the added edges for each split vertex. The expected cost of the added edges is exactly $\frac{1}{\rho \cdot Q} \sum_{i=1}^{k} 2 \cdot c(r, v_i)$.

We can derandomize the above procedure by enumerating all possible choices of the random offset s and output the cheapest solution among all possibilities. ∎

2 An Improved Approximation Algorithm for CVRP.

Recall that Λ denotes the number of tours in some reference optimal solution OPT, and $K := \frac{4 \cdot D}{\varepsilon} \cdot \lceil \frac{8 \cdot \pi \cdot D}{\varepsilon} \rceil$ denotes the number of cells in a cell-splitting of a D-bounded instance. Let $\Lambda' := \lceil \frac{2}{\varepsilon} \cdot K^2 \cdot D \rceil$. We remark that both K and D depend only on ε, hence $\Lambda' = O_\varepsilon(1)$. The next lemma summarizes some properties that we can assume w.l.o.g.

LEMMA 2.1. *For any two constants $\varepsilon > 0$ and $\alpha > 1$, given a polynomial-time α-approximation algorithm, $\alpha > 1$, for an instance of CVRP satisfying: (1) the instance is D-bounded, (2) $Q \ge 1/\varepsilon$, and (3) $\Lambda \ge \Lambda'$. Then there is a polynomial-time $\alpha(1+\varepsilon)$ approximation for CVRP in general.*

Proof. By Theorem 1.6 we can assume (1) while losing a factor $1 + \varepsilon$ in the approximation. If (2) does not hold, we obtain a $1 + \varepsilon' \le \alpha$ approximation using the PTAS in [15]. If (3) does not hold for some optimal solution, we obtain a $1 + \varepsilon' \le \alpha$ approximation using the PTAS in [4] when the number of tours is bounded by $\Lambda = O_\varepsilon(1)$, for each value $\Lambda \in \{\lceil n/Q \rceil, \dots, \Lambda'\}$. Otherwise, we use the α-approximation algorithm in the claim. The lemma follows. ∎

Therefore from now on we will assume that the input CVRP instance satisfies the properties (1)-(3) above. In order to lighten the notation, rather than using multiple ceelings and floors, we will rather assume that certain parameters defined later (e.g., $\rho \cdot Q$) are integer. This assumption increases the final approximation factor, but using the fact that Q is large enough, then this increase is negligible (and absorbed by other numerical roundings that we perform in the final computation).

Let OPT be an optimal solution for this instance whose cost is denoted by opt. By shortcutting, we can assume that each client belongs to *exactly one* tour in OPT. We define the cells and the respective quantities as in Section 1.3. All the constant parameters defined in this section will be set at the end of this section.

As stated in Section 1.2, when rlb is "far" from opt, then the existing techniques beats the factor 2 approximation. So it is sufficient to focus on instances where rlb is "close" to opt, that we next call *difficult* *instances.* In order to make the definition of difficult instances precise, we define the δ-neighboring cells of a center $z \in Z$ as follows, for a constant parameter $0 < \delta < 1$ to be fixed later.

DEFINITION 2.2 (δ-neighboring cells). *The δ-neighboring cells $\mathcal{N}_\delta(z)$ of $z \in Z$ are all the cells* $\text{cell}(z')$ *that contain at least one point at distance at most $\delta \cdot c(r,z) + \varepsilon$ from z and such that $c(r,z') \le c(r,z)$. See Figure 1.4 for an illustration.*

FACT 2.3. *Consider a* $\text{cell}(z)$*. The following holds:*

1. *For any point $x \in \mathbb{R}^2$ in a δ-neighboring cell of z and a point $y \in \mathbb{R}^2$ in* $\text{cell}(z)$*, we have $c(x,y) \le \delta \cdot c(r,z) + 2 \cdot \varepsilon$.*
2. *Let $x \in \mathbb{R}^2$ be a point such that $c(r,x) \le c(r,z)$, and x is* not *in any δ-neighboring cell of z. Then, x has distance at least $\delta \cdot c(r,z)$ to any point in* $\text{cell}(z)$*.*

Proof. Proof of (1): let x be a point in $\text{cell}(z') \in \mathcal{N}_\delta(z)$ and let v be a point in $\text{cell}(z')$ such that $c(z,v) \le \delta \cdot c(r,z) + \varepsilon$. Note that such v exists by the definition of δ-neighboring cells. Then, we can write $c(x,y) \le c(x,v) + c(v,z) + c(z,y) \le \frac{\varepsilon}{2} + (\delta \cdot c(r,z) + \varepsilon) + \frac{\varepsilon}{2}$, where the last inequality follows from Fact 1.7(1).

Proof of (2): since x is not in any δ-neighboring cell of z and $c(r,x) \le c(r,z)$, it must be that x is outside of the circle around z with radius $\delta \cdot c(r,z) + \varepsilon$. Therefore, the distance between x and any point in $\text{cell}(z)$ is at least $\delta \cdot c(r,z) + \varepsilon - \frac{\varepsilon}{2} \ge \delta \cdot c(r,z)$, again by Fact 1.7(1). ∎

Recall from Section 1.2 that, for a tour $T^* \in \text{OPT}$, $z(T^*)$ is the center of the cell containing the client $\text{peak}(T^*)$ of T^* which is farthest away from r. We distinguish between tours $T^* \in \text{OPT}$ depending on the number of clients outside the δ-neighboring cells of $z(T^*)$. Let $0 < \delta', \delta'' < 1$ be constant parameters that will be fixed later.

DEFINITION 2.4 (Bad and good tours). *Let $T^* \in \text{OPT}$. We call T^** bad*, if at least $\lceil \delta' \cdot Q \rceil$ many clients in $\mathcal{C}(T^*)$ do* not *belong to any δ-neighboring cell of $z(T^*)$. Otherwise, we call T^* a* good *tour.*

Let OPT_{good} (resp., OPT_{bad}) be the set of good (resp., bad) tours in OPT. The following lemma states that, if the total cost of bad tours in OPT is *large*, then the there is a *large* gap between rlb and opt.

LEMMA 2.5 (Bad tours imply easy instance). *If* $\sum_{T^* \in \text{OPT}_{\text{bad}}} c(T^*) \ge \delta'' \cdot \text{opt}$*, then*

$$\text{rlb} \le \left(1 - \frac{\delta \cdot (1-\delta) \cdot \delta' \cdot \delta''}{2}\right) \cdot \text{opt}.$$

Proof. For any bad tour $T^* \in \text{OPT}$, we show a gap between $\text{rlb}(\mathcal{C}(T^*))$ and $c(T^*)$ which proves the lemma. The following claim helps in establishing such gap.

CLAIM 2.6. *Let $T^* \in \text{OPT}$ be a bad tour. Let $v \in \mathcal{C}(T^*)$ be a client that does* not *belong to any δ-neighboring cell of $z(T^*)$. Then, we have*

$$2 \cdot c(r,v) \le \left(1 - \frac{\delta \cdot (1-\delta)}{2}\right) \cdot c(T^*). \tag{2.1}$$

Proof. We consider two cases:

Case 1: $c(r, z(T^*)) \le \frac{(1-\delta)}{2} \cdot c(T^*)$. In this case, we have

$$2 \cdot c(r,v) \le 2 \cdot c(r, \text{peak}(T^*)) \le 2 \cdot c(r, z(T^*)) \le (1-\delta) \cdot c(T^*),$$

where the second inequality follows from Fact 1.7 (2).

Case 2: $c(r, z(T^*)) > \frac{(1-\delta)}{2} \cdot c(T^*)$. Note that we have the following inequality for any $v \in \mathcal{C}(T^*)$:

$$2 \cdot c(r,v) + c(v, \text{peak}(T^*)) \le c(r,v) + c(v, \text{peak}(T^*)) + c(r, \text{peak}(T^*)) \le c(T^*). \tag{2.2}$$

Now we can write

$$\begin{aligned} 2 \cdot c(r,v) &\le c(T^*) - c(v, \text{peak}(T^*)) \\ &\le c(T^*) - \delta \cdot c(r, z(T^*)) \\ &\le c(T^*) - \delta \cdot \frac{1-\delta}{2} \cdot c(T^*), \end{aligned}$$

where the first inequality follows from (2.2), the second inequality holds by Fact 2.3(2) and the last inequality is true because of the lower bound on $c(r, z(T^*))$ as we are in Case 2.

This finishes the proof of the claim. ∎

We proceed with the proof of the lemma. We show for a tour $T^* \in \mathrm{OPT}_{\mathrm{bad}}$ there is a gap between $\mathrm{rlb}(\mathcal{C}(T^*))$ and $c(T^*)$ as follows:

$$\begin{aligned}\mathrm{rlb}(\mathcal{C}(T^*)) &= \frac{1}{Q}\sum_{v\in\mathcal{C}(T^*)} 2\cdot c(r,v)\\ &\le \frac{(1-\delta')\cdot Q}{Q}\cdot c(T^*) + \frac{\delta'\cdot Q}{Q}\cdot(1-\frac{\delta\cdot(1-\delta)}{2})\cdot c(T^*)\\ &= (1-\frac{\delta\cdot(1-\delta)\cdot\delta'}{2})\cdot c(T^*),\end{aligned}\tag{2.3}$$

where the inequality follows from the fact that there are at least $\lceil\delta'\cdot Q\rceil \ge \delta'\cdot Q$ many clients in $\mathcal{C}(T^*)$ such that (2.1) holds for them.

Finally, we can prove the lemma using the above gap between $c(T^*)$ and $\mathrm{rlb}(\mathcal{C}(T^*))$ for bad tours in OPT. Recall that, w.l.o.g., we could assume that a client belongs to exactly one tour in OPT. Hence, we can write

$$\begin{aligned}\mathrm{rlb} &= \sum_{T^*\in\mathrm{OPT}_{\mathrm{bad}}}\mathrm{rlb}(\mathcal{C}(T^*)) + \sum_{T^*\in\mathrm{OPT}_{\mathrm{good}}}\mathrm{rlb}(\mathcal{C}(T^*))\\ &\le \sum_{T^*\in\mathrm{OPT}_{\mathrm{bad}}}(1-\frac{\delta\cdot(1-\delta)\cdot\delta'}{2})\cdot c(T^*) + \sum_{T^*\in\mathrm{OPT}_{\mathrm{good}}} c(T^*)\\ &\le (1-\frac{\delta\cdot(1-\delta)\cdot\delta'\cdot\delta''}{2})\cdot\mathrm{opt},\end{aligned}$$

where the first inequality follows from (2.3) for bad tours and the fact that $\mathrm{rlb}(\mathcal{C}(T^*)) \le c(T^*)$ for any tour with at most Q clients (hence specifically for good tours). The second inequality follows since we are assuming $\sum\limits_{T^*\in\mathrm{OPT}_{\mathrm{bad}}} c(T^*) \ge \delta''\cdot\mathrm{opt}$. ∎

Now, we can formalize the definition of difficult instances.

Definition 2.7 (Difficult instances). *We say that the considered instance of CVRP is difficult if*

$$\mathrm{rlb} \ge (1-\frac{\delta\cdot(1-\delta)\cdot\delta'\cdot\delta''}{2})\cdot\mathrm{opt}.$$

In the following we summarize some useful properties of difficult instances. We note here that the inequalities *(ii)* and *(iii)* in Lemma 2.8 are far from being tight; however, this will suffice for our purposes.

Lemma 2.8 (Properties of difficult instances). *For difficult instances, we have*

(i) $\mathrm{regret}(\mathrm{OPT}) := \mathrm{opt} - \mathrm{plb} \le \frac{\delta\cdot(1-\delta)\cdot\delta'\cdot\delta''}{2}\cdot\mathrm{opt}$

(ii) $4\cdot\sum\limits_{z\in Z} c(r,z) \le \varepsilon\cdot\mathrm{opt}$

(iii) $2\cdot\sum\limits_{z\in Z}\mathrm{bd}(z) \le \varepsilon\cdot\mathrm{opt}.$

(iv) $\sum\limits_{T^*\in\mathrm{OPT}_{\mathrm{bad}}} c(T) \le \delta''\cdot\mathrm{opt}.$

Proof. *(i)* follows from (1.1) together with Definition 2.7 and the fact that rlb ≤ plb (see Lemma 1.4). *(ii)* follows directly from Properties (1) and (3) of Lemma 2.1. *(iii)* holds because Fact 1.7 says that $\mathrm{bd}(z) \le \varepsilon$ for any $z \in Z$, together with Property (3) of Lemma 2.1. *(iv)* follows from Lemma 2.5 together with Definition 2.7. ∎

Our final algorithm for the considered instance of CVRP is sketched in Algorithm 2.1.

Algorithm 2.1 Improved Algorithm for CVRP

Input: An instance of CVRP satisfying (1)-(3) in Lemma 2.1

Output: A feasible solution.

1: Compute a solution APX_1 with the algorithm from Lemma 1.3.

2: Compute a solution APX_2 with Algorithm 3.1 in Section 3 for a given parameter $0 < \rho < \frac{1}{2}$ where $\rho \cdot Q$ is integer.

3: Output the cheapest solution between APX_1 and APX_2.

If the input instance is not difficult, then $c(APX_1) \leq (2 + \varepsilon - \frac{\delta \cdot (1-\delta) \cdot \delta' \cdot \delta''}{2}) \cdot \text{opt}$ by Lemmas 1.3 and 1.4. Otherwise, namely if the input instance is difficult, the following lemma upper bounds the cost of APX_2.

LEMMA 2.9 (Final cost of Algorithm 3.1). *For any $0 < \delta, \delta', \delta'' < 1$, and any $0 < \rho < \frac{1}{2}$ where $\rho \cdot Q$ is an integer, the approximation factor of Algorithm 3.1 on a difficult instance is at most*

$$1 + 2 \cdot \delta + 2 \cdot \rho + \frac{\delta' + \delta''}{\rho} + \delta \cdot (1 - \delta) \cdot \delta' \cdot \delta'' + 21 \cdot \varepsilon$$

The proof of our main theorem follows easily, by just optimizing the constant parameters.

Proof of Theorem 1.2. By Lemma 2.1, by losing a factor $1 + \varepsilon$ in the approximation we can assume that properties (1)-(3) hold. Apply Algorithm 2.1. From the above discussion the approximation factor of the latter algorithm is at most

$$\max\{2 + \varepsilon - \frac{\delta \cdot (1 - \delta) \cdot \delta' \cdot \delta''}{2}, 1 + 2 \cdot \delta + 2 \cdot \rho + \frac{\delta' + \delta''}{\rho} + \delta \cdot (1 - \delta) \cdot \delta' \cdot \delta'' + 21 \cdot \varepsilon\}$$

Let us fix $\varepsilon = 10^{-10}$, $\delta := 0.09152463, \delta' := 0.04161686, \delta'' := 0.04177375$. Let also $\rho' := 0.20444372$, and define $\rho := \rho' + \frac{\lceil \rho' \cdot Q \rceil - \rho' \cdot Q}{Q}$. Note that $\rho \cdot Q$ is an integer as required. Furthermore, since $Q \geq \frac{1}{\varepsilon}$, $\rho - \rho' \leq \varepsilon$. The claimed value of $\tau > 0$ follows. In more detail, we either get a 1.99993 approximation if the instance is not difficult or a 1.99998 approximation if the instance is difficult. ∎

3 Dealing with Difficult Instances. In this section we present an approximation algorithm, namely Algorithm 3.1, for difficult instances (see Definition 2.7) with approximation factor better than 2 (specifically as in Lemma 2.9). We next describe in more detail the different parts of the algorithm.

3.1 Peak Clients Configuration. The algorithm starts by guessing an optimal *peak clients configuration*, which is defined as follows. Recall that $\mathcal{C}(z)$ denotes the set of clients in cell(z). For every cell(z) and every δ-neighboring cell cell(z') of z, we guess the number $n^z_{z'}$ of clients in cell(z') that are on a tour in OPT whose peak center is z. Recall that $\mathcal{N}_\delta(z)$ is the set of all δ-neighboring cells of z. For notational convenient we define $n^z_{z'} = 0$ if $z' \notin \mathcal{N}_\delta(z)$. Let $k_{z'} := \sum_{z \in Z} n^z_{z'}$ and $n^z := \sum_{z' \in Z} n^z_{z'}$. Intuitively, $k_{z'}$ is the number of clients in cell(z') that are on an optimal tour whose peak is "near" z', while n^z is the number of peak clients of tours with peak center z. We also guess the number t_z of tours $T^* \in$ OPT with peak center $z(T^*) = z$. Observe that $n^z \leq t_z Q$ by definition. We call such a tuple $\{n^z_{z'}, t_z\}_{z,z' \in Z}$ a peak client configuration or simply configuration. We define *valid* configurations as follows:

DEFINITION 3.1 (Valid configuration). *We say that a configuration $\{n^z_{z'}, t_z\}_{z,z' \in Z}$ is* **valid** *if for every $z \in Z$, we have*

(i) $n^z = \sum_{z' \in Z} n^z_{z'} \leq t_z \cdot Q$,

(ii) $n^z_{z'} = 0$ *if* cell$(z') \notin \mathcal{N}_\delta(z)$, *and*

(iii) $k_{z'} = \sum_{z \in Z} n^z_{z'} \leq |\mathcal{C}(z')|$.

The algorithm considers all the possible valid configurations (line 1). We remark that this is a polynomial-size set. For each valid configuration $\{n^z_{z'}, t_z\}_{z,z' \in Z}$, it performs the following steps, therefore computing a feasible solution APX.

Algorithm 3.1 Algorithm for dealing with difficult instances

Input: An instance of CVRP satisfying (1)-(3) in Lemma 2.1, and a parameter $0 < \rho < \frac{1}{2}$ where $\rho \cdot Q$ is integer
Output: A feasible solution.

for all possible valid configuration $\{n^z_{z'}, t_z\}_{z,z' \in Z}$ (see Section 3.1) **do**
  Let $APX \leftarrow \emptyset$.
  for all $z' \in Z$ **do**
    Compute a $k_{z'}$-TSP tour $T_{\text{k-tsp}}(z')$ over $\mathcal{C}(z')$, $k_{z'} = \sum_{z \in Z} n^z_{z'}$ (see Section 3.2).
  end for
  Let $\mathcal{C}_{peak} := \bigcup_{z' \in Z} \mathcal{C}(T_{\text{k-tsp}}(z'))$ and $\mathcal{C}_{lfto} := \mathcal{C} \setminus \mathcal{C}_{peak}$.
  for all $z \in Z$ **do**
    Based on the $\{T_{\text{k-tsp}}(z')\}_{z' \in Z}$, compute a tour $T_{\text{peak}}(z)$ containing $n^z = \sum_{z' \in Z} n^z_{z'}$ clients (see Section 3.2).
  end for
  Compute a minimum cost collection of trees $F_1, \ldots, F_q$, $q = \sum_{z \in Z} 2 \cdot t_z$, such that: (i) the trees span all the clients $\mathcal{C}_{lfto}$; (ii) each F_i contains r and some $z_i \in Z$; (iii) the number of trees with $z_i = z$ is exactly $2 \cdot t_z$ (see Section 3.3).
  for $i = 1, \ldots, q$ **do**
    Convert F_i into an r-z_i path P_i (see Section 3.4).
    Convert P_i into tours $T^0_i, \ldots, T^{h_i}_i$, each one containing r and at most $\rho \cdot Q$ clients, and an r-z_i path P'_i containing at most $\rho \cdot Q$ clients, such that all the clients of P_i belong to a tour T^j_i or to P'_i (see Section 3.4).
    Add $\{T^0_i, \ldots, T^{h_i}_i\}$ to APX.
  end for
  for all $z \in Z$ **do**
    Let $\mathcal{P}_z$ be the collection of $2 \cdot t_z$ paths P'_i with $z_i = z$. Pair them arbitrarily as $(P^{(a)}_{1,z}, P^{(b)}_{1,z}), \ldots, (P^{(a)}_{t_z,z}, P^{(b)}_{t_z,z})$.
    Partition $T_{\text{peak}}(z)$ into t_z fragments $T^1_{\text{peak}}(z), \ldots, T^{t_z}_{\text{peak}}(z)$ containing up to $(1 - 2 \cdot \rho) \cdot Q$ clients each, plus a minimal number $t'_z \geq 0$ of fragments $T^{t_z+1}_{\text{peak}}(z), \ldots, T^{t_z+t'_z}_{\text{peak}}(z)$ containing up to Q clients each (see Section 3.5).
    for $j = 1, \ldots, t_z$ **do**
      Combine $(P^{(a)}_{j,z}, P^{(b)}_{j,z})$ and $T^j_{\text{peak}}(z)$ into a tour T^j_z and add it to APX (see Section 3.5).
    end for
    for $j = t_z + 1, \ldots, t_z + t'_z$ **do**
      Convert $T^j_{\text{peak}}(z)$ into a tour T^j_z and add it to APX (see Section 3.5).
    end for
  end for
end for
Output the cheapest solution APX obtained above.

3.2 Computation of Tours on Peak Clients. Recall that a k-TSP tour over a set of nodes V, $|V| \geq k \geq 1$ is a tour visiting at least k nodes (chosen arbitrarily). In Line 4, for each $z' \in Z$, the algorithm computes a $k_{z'}$-TSP tour $T_{k-tsp}(z')$ over the clients $\mathcal{C}(z')$ in cell cell(z'). Such tour is computed using a known PTAS [3, 29]. We remark that, by the definition of a valid configuration, $|\mathcal{C}(z')| \geq k_{z'}$, hence this step is implementable. By $\mathcal{C}_{peak}$ (line 5) we denote the set of clients involved in such tours, that we next call *peak* clients. The remaining clients $\mathcal{C}_{lfto} := \mathcal{C} \setminus \mathcal{C}_{peak}$ are next called *leftover* clients.

Then we use fragments of the tours $T_{\text{k-tsp}}(z')$ to build a tour $T_{\text{peak}}(z)$ for each center z, containing n^z distinct clients in $\mathcal{C}_{peak}$ (Line 7). First of all, we break $T_{\text{k-tsp}}(z')$ into one fragment containing $n^z_{z'}$ distinct consecutive clients for each $z \in Z$. When $n^z_{z'} = 0$, the fragment is simply a dummy fragment containing no node. Notice that the fragments span precisely the $k_{z'}$ nodes of $T_{\text{k-tsp}}(z')$. Next for each $z \in Z$ we build $T_{\text{peak}}(z)$ by considering, for each $z' \in Z$ such that $n^z_{z'} > 0$, a distinct fragment of $T_{\text{k-tsp}}(z')$ with exactly $n^z_{z'}$ clients, and forming a tour including such fragments plus a minimal number of edges linking the endpoints of each fragment to form a tour in an arbitrary way (as we will see, there is no need to optimize here).

3.3 Covering the Leftover Clients via a Forest. We next show how to cover the leftover clients with a cheap forest via matroid intersection (Line 8). As mentioned in the introduction, this is the most novel part of our construction. Recall that our goal is to construct a forest of (not disjoint) trees $F_1, \ldots, F_q$ of total minimum cost such that:

(i) Each $v \in \mathcal{C}_{lfto}$ belongs to exactly one tree F_i;

(ii) Each F_i contains r and some center z_i;

(iii) There are precisely $2 \cdot t_z$ many trees having $z_i = z$.

Make $2 \cdot t_z$ copies of z collocated with z for each $z \in Z$, and let X be the set of all such copies. Similarly, make $\sum_{z \in Z} 2 \cdot t_z$ copies of r collocated with r, and let R be the set containing all the copies of r. Let $V' := \mathcal{C}_{lfto} \cup R \cup X$, and let $E' = \binom{V'}{2}$ be the set of pairs of distinct nodes in V'. We define a first matroid $M_1 = (E', I_1)$ where I_1 is the set of all the forests in (V', E') such that each connected component has at most one vertex in X. We define a second matroid $M_2 = (E', I_2)$, where I_2 is the set of all the forests in (V', E') such that each connected component has at most one vertex in R. The proof that M_1 and M_2 are matroids is similar to an analogous proof in [23]. For completeness, we include a complete proof of these claims in Appendix A. Any basis in the intersection of these two matroids is a forest that has exactly $|R| = |X|$ many trees, each one containing exactly one copy of r and one copy of some center z. Hence this induces a feasible solution to our problem (after collapsing back all the copies of r and of the centers). Using a weighted matroid intersection algorithm (e.g., Theorem 41.6 in [26]), we find a minimum cost such basis, that forms our forest $F_1, \ldots, F_q$.

3.4 Converting Trees into Tours and Paths. We next take each tree F_i in the forest (containing r and some center z_i), and convert it into an r-z_i path P_i covering the same set of leftover clients (Line 10). To do so we consider the unique r-z_i path $\tilde{P}_i$ in F_i, and duplicate all the edges in $E(F_i) \setminus E(\tilde{P}_i)$. Then we construct an Euler walk from r to z_i in the obtained multigraph and finally shortcut duplicated nodes.

Next we discuss how to implement Line 11. Here we simply apply the path partitioning Lemma 1.8. Notice that the obtained tours $T_i^0, \ldots, T_i^{h_i}$ are valid tours (i.e., they contain r and at most $\rho \cdot Q < Q$ clients each).

3.5 Covering Peak Clients and the Remaining Leftover Clients. We next discuss how to construct the final tours added in Lines 15, 17 and 19. These tours cover the leftover clients in the tours $P'_1, \ldots, P'_q$ plus all the peak clients $\mathcal{C}_{peak}$.

We do the following for each center $z \in Z$. First of all (Line 14), we subdivide $T_{\text{peak}}(z)$ into fragments (again formed by consecutive distinct clients) as follows. Remove any edge from $T_{\text{peak}}(z)$ and let T' be the resulting path. Iteratively remove the first $\min\{(1 - 2 \cdot \rho) \cdot Q, |\mathcal{C}(T')|\}$ clients from T', until all the clients of T' are removed or we have already identified t_z fragments. If the number of identified fragments is strictly smaller than t_z, we add a sufficient number of dummy fragments consisting of z only to reach exactly t_z fragments (the dummy fragments are only introduced to simplify the later notation). Let $T^1_{\text{peak}}(z), \ldots, T^{t_z}_{\text{peak}}(z)$ be the obtained fragments. If there are no remaining clients in T' we are done. Otherwise, iteratively remove the first $\min\{Q, |\mathcal{C}(T')|\}$ clients from T' until all the clients are removed from T'. Let $T^{t_z+1}_{\text{peak}}(z), \ldots, T^{t_z+t'_z}_{\text{peak}}(z)$ be the obtained fragments. Notice that possibly $t'_z = 0$, i.e., there are no such final fragments.

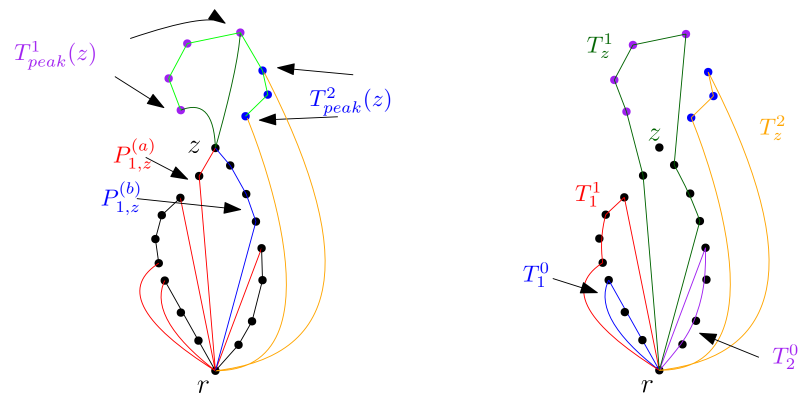

Figure 3.1: In this example, $t_z = t'_z = 1$. On the left, $T_{\text{peak}}(z)$ (line 7) is shown in light green color. The fragment $T^1_{\text{peak}}(z)$ (line 15) is shown with purple nodes. The fragment $T^2\text{peak}(z)$ (line 15) is shown with blue nodes. According to Section 3.5, we connect the endpoints of $T^1_{\text{peak}}(z)$ (dark green) and the endpoints of $T^2_{\text{peak}}(z)$ to r (orange). Since $|\mathcal{P}_z| = 2$, we only have one pair of paths $(P^{(a)}_{1,z}, P^{(b)}_{1,z})$. On the right, we show the resulting tours added to APX. Tours T^0_1, T^1_1, T^0_2 are added to APX in line 12. Tour T^1_z is added to APX in line 17 and tour T^2_z is added to APX in line 19.

The next steps are implemented trivially. For $j = 1, \ldots, t_z$, in Line 17 we construct a valid tour T^j_z (which is added to APX) as follows. We take the union of the pair of paths $(P^{(a)}_{j,z}, P^{(b)}_{j,z})$, plus the fragment $T^j_{\text{peak}}(z)$, plus two edges between the endpoints of the considered fragment and z. Then we shortcut z. Notice that the obtained tour contains r and all the clients in $P^{(a)}_{j,z}$, $P^{(b)}_{j,z}$, and $T^j_{\text{peak}}(z)$. Notice also that this tour is valid since $P^{(a)}_{j,z}$ and $P^{(b)}_{j,z}$ contain at most $\rho \cdot Q$ clients each, while $T^j_{\text{peak}}(z)$ contains at most $(1 - 2 \cdot \rho) \cdot Q$ clients.

For $j = t_z + 1, \ldots, t_z + t'_z$, in Line 19 we construct a valid tour T^j_z (which is added to APX) by adding to the fragment $T^j_{\text{peak}}(z)$ (which contains at most Q clients) one edge between r and each one of its two endpoints. Clearly this is a valid tour.

Notice that at this point all clients belong to at least one tour in APX and recall that each such tour contains at most Q clients. Hence APX is a feasible solution. The very last step of the algorithm is to return the cheapest solution APX among all the solutions computed for each valid peak clients configuration. See Figure 3.1 for an illustration of this algorithm.

4 Cost Analysis of Algorithm 3.1 and Proof of Lemma 2.9. In this section, we analyze the cost of the solution produced by Algorithm 3.1 which in turn proves Lemma 2.9. Consider the *optimal configuration* $\{n^z_{z'}, t_z\}_{z,z' \in Z}$ corresponding to OPT. Let APX be the solution computed given the optimal configuration. We bound $c(APX)$ which is an upper bound on the cost of the output solution. Our plan in this section is to bound the cost of tours added to APX in line 12, line 17, and line 19 separately. All the referred lines are based on Algorithm 3.1.

4.1 Cost of tours added in line 12. The tours T^j_i's and paths P'_i's (line 11) are obtained by applying Lemma 1.8 to P_i for $i = 1, \ldots, q$ (line 11). First we bound the cost of P_i's (computed in line 11).

Claim 4.1 (Cost of paths in line 11).

$$\sum_{i=1}^{q} c(P_i) \leq (1 + \varepsilon) \cdot \text{opt} + \text{regret(OPT)}.$$

Proof. Recall P_i is obtained from tree F_i for $i = 1, \ldots, q$ by doubling all the edges in F_i except the edges on the unique r-z_i path $\tilde{P}_i$ in F_i. Since $c(\tilde{P}_i) \geq c(r, z_i)$, we can write

$$\begin{aligned}
\sum_{i=1}^{q} c(P_i) &\leq \sum_{i=1}^{q} c(F_i) + (\sum_{i=1}^{q} c(F_i) - \sum_{i=1}^{q} c(r, z_i)) \\
&= 2 \cdot \sum_{i=1}^{q} c(F_i) - \sum_{z \in Z} 2 \cdot t_z \cdot c(r, z) \\
&\leq 2 \cdot \sum_{i=1}^{q} c(F_i) - \sum_{T^* \in \mathrm{OPT}} 2 \cdot c(r, \mathrm{peak}(T^*)) \\
&= 2 \cdot \sum_{i=1}^{q} c(F_i) - \mathrm{plb},
\end{aligned} \tag{4.1}$$

where the first equality holds because there are exactly $2 \cdot t_z$ many trees F_i's with $z_i = z$, the second inequality follows from the fact that there are t_z many tours in OPT whose peaks are in cell(z) and the fact that $c(r, z)$ is at least $c(r, v)$ for any $v \in \mathcal{C}(z)$ (see Fact 1.7(2)), specifically $c(r, z(T^*)) \geq c(r, \mathrm{peak}(T^*))$.

The claim follows by showing $\sum_{i=1}^{q} c(F_i) \leq (1 + \frac{\varepsilon}{2}) \cdot \mathrm{opt}$. This is easy to see: for each tour T^* in OPT add two edges $\{\mathrm{peak}(T^*), z(T^*)\}$. Transform T^* and these two added edges to a tour T' that contains $z(T^*)$ by shortcutting pass peak(T^*) once. Define peak(T') to be $z(T^*)$. Let OPT$'$ be the resulting tours. It is easy to see OPT$'$ induces a basis in both matroids M_1 and M_2 (see Section 3.3) by viewing each tour T' in OPT$'$ as two edge-disjoin r-peak(T') paths. Note that the cost of added edges for each tour in OPT is at most $2 \cdot \frac{\varepsilon}{2}$ because of the cell sizes (see Fact 1.7(1)). Therefore, $\sum_{i=1}^{q} c(F_i) \leq c(\mathrm{OPT}') \leq \mathrm{opt} + \sum_{T^* \in \mathrm{OPT}} 2 \cdot \frac{\varepsilon}{2} \leq (1 + \frac{\varepsilon}{2}) \cdot \mathrm{opt}$, where the last inequality holds because each tour in OPT has cost at least 2 (by definition of D-boundedness in Section 1.3). Plugging this bound for F_i's in (4.1) finishes the proof. ∎

Equipped with the above claim, we now bound the cost of tours in APX added in line 12 of the algorithm.

Lemma 4.2 (Cost of tours added in line 12).

$$\sum_{i=1}^{q} \sum_{j=0}^{h_i} c(T_i^j) \leq (1 + \frac{\delta' + \delta''}{\rho} + 2 \cdot \varepsilon) \cdot \mathrm{opt} + \mathrm{regret(OPT)} - \sum_{i=1}^{q} c(P'_i).$$

Proof. The tours T_i^j's and paths P'_i's (line 11) are obtained by applying Lemma 1.8 to P_i for $i = 1, \ldots, q$ (line 11). Note $\bigcup_{i=1}^{q} \mathcal{C}(P_i) = \mathcal{C}_{lfto}$. Hence, by the bound given in Lemma 1.8 for each P_i we can write:

$$\begin{aligned}
\sum_{i=1}^{q} \sum_{j=0}^{h_i} c(T_i^j) + \sum_{i=1}^{q} c(P'_i) &\leq \sum_{i=1}^{q} c(P_i) + \frac{1}{\rho \cdot Q} \sum_{v \in \mathcal{C}_{lfto}} 2 \cdot c(r, v) \\
&\leq (1 + \varepsilon) \cdot \mathrm{opt} + \mathrm{regret(OPT)} + \frac{1}{\rho \cdot Q} \sum_{v \in \mathcal{C}_{lfto}} 2 \cdot c(r, v),
\end{aligned} \tag{4.2}$$

where the second inequality follows from Claim 4.1. Below, we show $\frac{1}{\rho \cdot Q} \sum_{v \in \mathcal{C}_{lfto}} 2 \cdot c(r, v) \leq (\frac{\delta' + \delta''}{\rho} + \frac{\varepsilon}{2}) \cdot \mathrm{opt}$. Substituting this bound in (4.2) and moving $\sum_{i=1}^{q} c(P'_i)$ to the other side finishes the proof of the lemma.

Recall the definition of good tours in OPT in Definition 2.4. We say a client v is a *good client*, if v is on a good tour T^* and v belongs to a δ-neighboring cell of $z(T^*)$. The rest of the clients are called *bad clients*.

Consider cell cell(z') and a good client $v \in \mathcal{C}(z')$. Since v is a good client, there must be a tour $T^* \in \mathrm{OPT}_{\mathrm{good}}$ such that $v \in \mathcal{C}(T^*)$ and cell$(z') \in \mathcal{N}_\delta(z(T^*))$. Therefore, v contributed to $n_{z'}^{z(T^*)}$. We conclude $|\mathcal{C}_{peak} \cap \mathcal{C}(z')| = k_{z'} = \sum_{z \in Z} n_{z'}^{z} \geq |\mathcal{C}_{\mathrm{good}} \cap \mathcal{C}(z')|$ which in turn implies $|\mathcal{C}_{lfto} \cap \mathcal{C}(z')| \leq |\mathcal{C}_{\mathrm{bad}} \cap \mathcal{C}(z')|$. Together with the fact that distances from r to the clients in cell(z') differ from each other by an additive term $\frac{\varepsilon}{2}$ (i.e., $|c(r,v)-c(r,u)| \le \frac{\varepsilon}{2}$ for all $u,v \in \mathcal{C}(z')$, see Fact 1.7(1)) we have

$$\begin{aligned}
\frac{1}{Q}\cdot \sum_{v\in\mathcal{C}_{lfto}} 2\cdot c(r,v) &\le \frac{1}{Q}\cdot \sum_{z'\in Z}\sum_{u\in\mathcal{C}_{\text{bad}}\cap\mathcal{C}(z')} 2\cdot (c(r,u)+\frac{\varepsilon}{2}) \\
&= \frac{1}{Q}\cdot \sum_{u\in\mathcal{C}_{\text{bad}}} 2\cdot (c(r,u)+\frac{\varepsilon}{2}) \\
&= \frac{1}{Q}\cdot \sum_{u\in\mathcal{C}_{\text{bad}}} 2\cdot c(r,u) + \frac{\varepsilon}{2\cdot Q}\cdot \sum_{u\in\mathcal{C}_{\text{bad}}} 2 \\
&\le \frac{1}{Q}\cdot \sum_{u\in\mathcal{C}_{\text{bad}}} 2\cdot c(r,u) + \frac{\varepsilon}{2}\cdot \text{rlb} \\
&= \frac{1}{Q}\cdot \Big(\sum_{T^*\in\text{OPT}_{\text{good}}} \sum_{\substack{u\in\\ \mathcal{C}(T^*)\cap\mathcal{C}_{\text{bad}}}} 2\cdot c(r,u) \\
&+ \sum_{T^*\in\text{OPT}_{\text{bad}}}\sum_{u\in\mathcal{C}(T^*)} 2\cdot c(r,u)\Big) + \frac{\varepsilon}{2}\cdot\text{rlb} \\
&\le \frac{1}{Q}\cdot \Big(\delta'\cdot Q\cdot \sum_{T^*\in\text{OPT}_{\text{good}}} c(T^*) + Q\cdot \sum_{T^*\in\text{OPT}_{\text{bad}}} c(T^*)\Big) + \frac{\varepsilon}{2}\cdot\text{rlb} \\
&\le (\delta'+\delta''+\frac{\varepsilon}{2})\cdot\text{opt},
\end{aligned}$$

where the first inequality follows from the discussion above, and the second inequality follows from the definition of rlb and the fact that distance of any client from r is at least 1. The penultimate inequality holds since $2\cdot c(r,u) \le c(T^*)$ for all $u\in\mathcal{C}(T^*)$ for any tour in OPT, each good tour has at most $\delta'\cdot Q$ bad clients and bad tours have at most Q clients. The last inequality follows from Lemma 2.8(iv) that states $\sum_{T^*\in\text{OPT}_{\text{bad}}} c(T^*) \le \delta''\cdot\text{opt}$ and the fact that rlb $\le$ opt. ∎

4.2 Cost of tours added in lines 17 and 19. Next we compute the cost of tours added to APX in lines 17 and 19 of the algorithm. Recall for center z, $T_{peak}(z)$ is partitioned into fragments $T^1_{\text{peak}}(z),\dots,T^{t_z}_{\text{peak}}(z)$ (which will be part of the tours T^j_z for $j=1,\dots,t_z$) and fragments $T^{t_z+1}_{\text{peak}}(z),\dots,T^{t_z+t'_z}_{\text{peak}}(z)$ (which will be part of the tours T^j_z for $j=t_z+1,\dots,t_z+t'_z$).

Lemma 4.3 (Cost of tours added in line 17).

$$\sum_{z\in Z}\sum_{j=1}^{t_z} c(T^j_z) \le \sum_{i=1}^{q} c(P'_i) + \sum_{z\in Z}\sum_{j=1}^{t_z} c(T^i_{\text{peak}}(z)) + \delta\cdot\text{plb} + 3\cdot\varepsilon\cdot\text{opt}.$$

Proof. Consider cell cell(z) and a tour T^j_z for some $1\le j\le t_z$. Note T^j_z is obtained from the j-th fragment $T^j_{\text{peak}}(z)$ and the j-th pair of paths $(P^{(a)}_{j,z},P^{(b)}_{j,z})$ by adding an edge between z and each of the two endpoints of $T^j_{\text{peak}}(z)$ (see Section 3.5). Note each added edge is between z and a client in a δ-neighboring cell of z which has cost at most $\delta\cdot c(r,z)+2\cdot\varepsilon$ by Fact 2.3(1). Since we add two such edges to build T^j_z, we get

$$c(T^j_z) \le c(P^{(a)}_{j,z}) + c(P^{(b)}_{j,z}) + c(T^j_{\text{peak}}(z)) + 2\cdot(\delta\cdot c(r,z)+2\cdot\varepsilon).$$

Summing the above inequality for all T^j_z for $j=1,\dots,t_z$ we get (4.3). Recall that $\mathcal{P}_z$ (line 14) is the collection of paths P'_i with $z_i = z$ for $i=1,\dots,q$.

$$\begin{aligned}\sum_{j=1}^{t_z} c(T_z^j) &\leq \sum_{j=1}^{t_z}\left(c(P_{j,z}^{(a)}) + c(P_{j,z}^{(b)}) + c(T_{\text{peak}}^j(z)) + \delta\cdot 2\cdot c(r,z) + 4\cdot\varepsilon\right)\\ &= \sum_{P\in\mathcal{P}_z} c(P) + \sum_{j=1}^{t_z} c(T_{\text{peak}}^j(z)) + \delta\cdot 2\cdot t_z\cdot c(r,z) + 4\cdot t_z\cdot\varepsilon\end{aligned} \tag{4.3}$$

By summing the above inequality for all $z \in Z$ we get the desired bound for the cost of the tours added to APX in line 17 where this calculation is shown below. Note $\sum_{z\in Z}\sum_{P\in\mathcal{P}_z} c(P) = \sum_{i=1}^{q} c(P'_i)$.

$$\begin{aligned}\sum_{z\in Z}\sum_{j=1}^{t_z} c(T_z^j) &\leq \sum_{z\in Z}\sum_{P\in\mathcal{P}_z} c(P) + \sum_{z\in Z}\sum_{j=1}^{t_z} c(T_{\text{peak}}^j(z)) + \delta\cdot\sum_{z\in Z} 2\cdot t_z\cdot c(r,z) + 2\cdot\varepsilon\sum_{z\in Z} 2\cdot t_z\\ &\leq \sum_{i=1}^{q} c(P'_i) + \sum_{z\in Z}\sum_{i=1}^{t_z} c(T_{\text{peak}}^j(z)) + \delta\cdot\sum_{T^*\in\text{OPT}} 2\cdot(c(r,\text{peak}(T^*)) + \frac{\varepsilon}{2}) + 2\cdot\varepsilon\sum_{z\in Z} 2\cdot t_z\\ &= \sum_{i=1}^{q} c(P'_i) + \sum_{z\in Z}\sum_{i=1}^{t_z} c(T_{\text{peak}}^j(z)) + \delta\cdot\text{plb} + \frac{\delta\cdot\varepsilon}{2}\sum_{T^*\in\text{OPT}} 2 + 2\cdot\varepsilon\sum_{z\in Z} 2\cdot t_z\\ &\leq \sum_{i=1}^{q} c(P'_i) + \sum_{z\in Z}\sum_{i=1}^{t_z} c(T_{\text{peak}}^j(z)) + \delta\cdot\text{plb} + \frac{\varepsilon}{2}\cdot\text{opt} + 2\cdot\varepsilon\cdot\text{opt},\end{aligned}$$

where the first inequality follows from (4.3), the second inequality holds because OPT has exactly t_z many tours whose peak is in cell(z) and the fact that $c(r, z(T^*)) \leq c(r, \text{peak}(T^*)) + \frac{\varepsilon}{2}$ (see Fact 1.7(1)), the equality follows from the definition of plb. The last inequality follows from the fact that OPT has t_z many tours of cost at least 2 for each $z \in Z$. ∎

Next, we bound the cost of tours T_z^j's for $j = t_z + 1, \ldots, t_z + t'_z$. Note t'_z might be 0 for some $z \in Z$ which in this case no tour is dded to APX in line 19 for center z.

LEMMA 4.4 (Cost of tours added in line 19).

$$\sum_{z\in Z}\sum_{j=t_z+1}^{t_z+t'_z} c(T_z^j) \leq \sum_{z\in Z}\sum_{j=t_z+1}^{t_z+t'_z} c(T_{\text{peak}}^j(z)) + 2\cdot\rho\cdot\text{plb} + \varepsilon\cdot\text{opt}.$$

Proof. For a cell z where $t'_z = 0$ no tour is added to APX. So consider a cell z with $t'_z \geq 1$. Let T_z^j be a tour added to APX for some $t_z + 1 \leq j \leq t_z + t'_z$. Recall T_z^j is obtained from the j-th fragment $T_{\text{peak}}^j(z)$ and two added edges: edge between r and each one of the two endpoints of $T_{\text{peak}}^j(z)$ (see Section 3.5). Note each added edge is between r and a client in a δ-neighboring cell of z. Any point in a δ-neighboring cell of z has cost at most $c(r, z)$ from r. Since we add two such edges to build T_z^j, we get

$$c(T_z^j) \leq c(T_{\text{peak}}^j(z)) + 2\cdot c(r,z).$$

Adding the above inequality for all the centers give

$$\sum_{j=t_z+1}^{t_z+t'_z} c(T_z^j) \leq \sum_{j=t_z+1}^{t_z+t'_z} c(T_{\text{peak}}^j(z)) + 2\cdot t'_z\cdot c(r,z). \tag{4.4}$$

Next, we give an upper bound on t'_z. Note $|\mathcal{C}(T_{\text{peak}}(z))| \leq t_z \cdot Q$ and the first t_z fragments of $T_{\text{peak}}(z)$ each has exactly $(1 - 2\cdot\rho)\cdot Q$ clients (because $t'_z \geq 1$). Therefore, the total number of clients on the remaining fragments

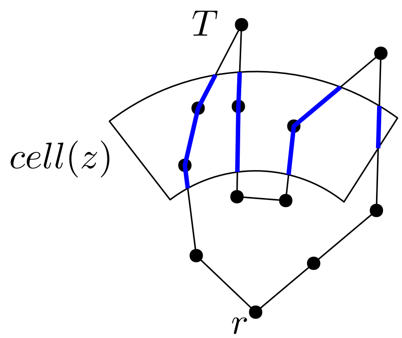

Figure 4.1: The projection of the tour T into cell(z) is shown in blue color. The nodes are clients.

is $|\bigcup_{j=t_z+1}^{t_z+t'_z} \mathcal{C}(T^j_{\text{peak}}(z))| \leq 2 \cdot \rho \cdot t_z \cdot Q$. Since each $T^j_{\text{peak}}(z)$ for $j = t_z+1, \ldots, t_z+t'_z$ has exactly Q clients except possibly the last one, we conclude $t'_z \leq \lceil 2 \cdot \rho \cdot t_z \rceil \leq 2 \cdot \rho \cdot t_z + 1$. Plugging this bound for t'_z into (4.4) we get

$$\sum_{j=t_z+1}^{t_z+t'_z} c(T^j_z) \leq \sum_{j=t_z+1}^{t_z+t'_z} c(T^j_{\text{peak}}(z)) + (2 \cdot \rho \cdot t_z + 1) \cdot 2 \cdot c(r, z).$$

By summing the above inequality for all $z \in Z$ we get the desired bound for the cost of the tours added to APX in line 19:

$$\begin{aligned}
\sum_{z \in Z} \sum_{j=t_z+1}^{t_z+t'_z} c(T^j_z) &\leq \sum_{z \in Z} \sum_{j=t_z+1}^{t_z+t'_z} c(T^j_{\text{peak}}(z)) + 2 \cdot \rho \cdot \sum_{z \in Z} 2 \cdot t_z \cdot c(r, z) + 2 \cdot \sum_{z \in Z} c(r, z) \\
&\leq \sum_{z \in Z} \sum_{j=t_z+1}^{t_z+t'_z} c(T^j_{\text{peak}}(z)) + 2 \cdot \rho \cdot \sum_{T^* \in \text{OPT}} 2 \cdot (c(r, \text{peak}(T^*)) + \frac{\varepsilon}{2}) + 2 \cdot \sum_{z \in Z} c(r, z) \\
&= \sum_{z \in Z} \sum_{j=t_z+1}^{t_z+t'_z} c(T^j_{\text{peak}}(z)) + 2 \cdot \rho \cdot \text{plb} + \rho \cdot \varepsilon \cdot \sum_{T^* \in \text{OPT}} 2 + 2 \cdot \sum_{z \in Z} c(r, z) \\
&\leq \sum_{z \in Z} \sum_{j=t_z+1}^{t_z+t'_z} c(T^j_{\text{peak}}(z)) + 2 \cdot \rho \cdot \text{plb} + \frac{\varepsilon}{2} \cdot \text{opt} + \frac{\varepsilon}{2} \cdot \text{opt}, \qquad \square
\end{aligned}$$

where the second inequality follows from the fact that OPT has exactly t_z tours with peak center z and $c(r, z(T^*)) \leq c(r, \text{peak}(T^*)) + \frac{\varepsilon}{2}$ (just like in the proof of Lemma 4.3), and the last inequality follows from the fact that $\rho < \frac{1}{2}$, each tour in OPT has cost at least 2 and Lemma 2.8(ii).

As it is clear from the statements of the previous two lemmas, we need to analyze $\sum_{z \in Z} c(T_{\text{peak}}(z))$ to finish off the cost analysis for this subsection. To do so, first we need to define the cost of a tour inside a cell. Consider a tour $T^* \in \text{OPT}$, and cell cell(z) drawn on the plane. Note T^* is a polygonal chains. Consider the projection of T^* into cell(z) that produces polygonal chains $\Pi_1, \ldots \Pi_k$ of T^* inside cell(z) where the endpoints of each Π_i for $i = 1, \ldots k$ are on the boundary of cell(z). Recall the definition of subpaths from Section 1.3 which can be applied to Π's as well. We define the cost of T^* inside cell(z) to be $c(T^*[\text{cell}(z)]) := \sum_{i=1}^{k} c(\Pi_i)$. See Figure 4.1 for an illustration.

In the next claim, we show on average the cost of optimal tours inside the δ-neighbroing cells of their peaks is small. This will be crucial later.

Claim 4.5 (Cost of a tour close to its peak). *Let T^* be a tour in* OPT*, and let P_{T^*}, P'_{T^*} be the corresponding*

r-peak(T^*) *path when viewing* T^* *as two paths between* r *and* peak(T^*)*. Then,*

$$\sum_{z':\ \mathrm{cell}(z')\in\mathcal{N}_\delta(z(T^*))} c(T^*[\mathrm{cell}(z')]) \le \delta\cdot 2\cdot c(r,\mathrm{peak}(T^*)) + \mathrm{regret}(P_{T^*}) + \mathrm{regret}(P'_{T^*}) + 5\cdot\varepsilon. \tag{4.5}$$

Proof. Consider the ordered sequence $r = v_0, v_1, \ldots, v_k$ of nodes along P_{T^*} where $v_i \in \mathcal{C}$ for $i = 1,\ldots,k$. Let $x \in \mathbb{R}^2$ be the point closest to v_i in the intersection of the line segment between $\{v_i, v_{i+1}\}$ and the boundary of a δ-neighboring cell of $z(T^*)$ where i is the smallest such index. Let $P^1_{T^*}$ be the subpath of P_{T^*} starting at r and ending at x, and let $P^2_{T^*}$ be the subpath of P_{T^*} starting at x and ending at peak(T^*) (see the Preliminary for the definition of these subpaths and their costs). Similarly, we define the $x' \in \mathbb{R}^2$ on P'_{T^*} and its corresponding subpaths $P'^1_{T^*}$ and $P'^2_{T^*}$. Note that the LHS of (4.5) is at most $c(P^2_{T^*}) + c(P'^2_{T^*})$.

Using Fact 1.5 about the regret of paths we write

$$\begin{aligned}\mathrm{regret}(P_{T^*}) &= c(P_{T^*}) - c(r,\mathrm{peak}(T^*))\\ &= c(P^1_{T^*}) + c(P^2_{T^*}) - c(r,\mathrm{peak}(T^*)).\end{aligned}$$

By rearranging the above equality, we get $c(P^2_{T^*}) = c(r,\mathrm{peak}(T^*)) - c(P^1_{T^*}) + \mathrm{regret}(P_{T^*})$. Since $c(P^1_{T^*}) \ge c(r,x)$, we conclude $c(P^2_{T^*}) \le c(r,\mathrm{peak}(T^*)) - c(r,x) + \mathrm{regret}(P_{T^*})$. Because x belongs to a δ-neighboring cell of $z(T^*)$ and peak(T^*) is in cell$(z(T^*))$, by Fact 2.3(1) and the triangle inequality we have

$$c(P^2_{T^*}) \le \delta\cdot c(r,z(T^*)) + 2\cdot\varepsilon + \mathrm{regret}(P_{T^*}).$$

Since $c(r,z(T^*)) \le c(r,\mathrm{peak}(T^*)) + \frac{\varepsilon}{2}$ (Fact 1.7(1)) we conclude the following bound for the cost of $P^2_{T^*}$:

$$c(P^2_{T^*}) \le \delta\cdot c(r,\mathrm{peak}(T^*)) + 2.5\cdot\varepsilon + \mathrm{regret}(P_{T^*}),$$

where we used the fact that $\delta\cdot\frac{\varepsilon}{2} \le \frac{\varepsilon}{2}$. The claim follows by obtaining a similar bound for $P'^2_{T^*}$. ◻

Finally, we are ready to bound the cost of $T_{\mathrm{peak}}(z)$'s.

LEMMA 4.6 (Cost of peak-tours).

$$\sum_{z\in Z} c(T_{\mathrm{peak}}(z)) \le \delta\cdot\mathrm{plb} + \mathrm{regret}(\mathrm{OPT}) + 15\cdot\varepsilon\cdot\mathrm{opt}.$$

Proof. Recall each tour $T_{\mathrm{peak}}(z)$ is obtained from some fragments of $T_{\text{k-tsp}}(z')$ for cell$(z') \in \mathcal{N}_\delta(z)$ and the added edges between the endpoints of these fragments (see Section 3.2). Note for each z, we have up to K (the number of cells) fragments and we can connect these fragments by at most K edges each of length $2\cdot D$ (because every point is in a ball of radius D). Therefore, we have

$$\begin{aligned}\sum_{z\in Z} c(T_{\mathrm{peak}}(z)) &\le \sum_{z'\in Z} c(T_{\text{k-tsp}}(z')) + 2\cdot\sum_{z'\in Z} K\cdot D\\ &\le \sum_{z'\in Z} c(T_{\text{k-tsp}}(z')) + \frac{\varepsilon}{2}\cdot\mathrm{opt},\end{aligned}\tag{4.6}$$

where the second inequality holds because each tour in OPT has cost at least 2; hence opt $\ge 2\cdot\Lambda' \ge \varepsilon\cdot(4\cdot K^2\cdot D)$ (see Definition 2.7(b)). So it remains to bound $\sum\limits_{z'\in Z} c(T_{\text{k-tsp}}(z'))$.

CLAIM 4.7. $\sum\limits_{z'\in Z} c(T_{\textit{k-tsp}}(z')) \le \delta\cdot\mathrm{plb} + \mathrm{regret}(\mathrm{OPT}) + 14\cdot\varepsilon\cdot\mathrm{opt}$

Proof. We use a similar fact proven in [18] that states one could turn a collection of paths (polygonal chains in our case) inside a cell whose endpoints lie on the boundary of the cell into a TSP tour by paying an extra $\frac{3}{2}$ times the length of the boundary of the cell. This was used in [14] as well. For cell(z'), let $\mathcal{T}^*_{z'}$ be the collection of all the tours $T^* \in$ OPT where cell$(z') \in \mathcal{N}_\delta(z(T^*))$. Note $k_{z'} = |\mathcal{C}(z') \cap (\bigcup\limits_{T^*\in\mathcal{T}^*_{z'}} \mathcal{C}(T^*))|$. Consider all the polygonal chains obtained from projection of T^* inside cell(z') for all $T^* \in \mathcal{T}^*_z$. These polygonal chains altogether cover exactly $k_{z'}$ clients. Using the known PTAS for k-TSP and the result of Karp, we conclude

$$c(T_{\text{k-tsp}}(z')) \leq (1+\varepsilon) \cdot \Big(\big(\sum_{T^* \in \mathcal{T}^*_{z'}} c(T^*[\text{cell}(z')])\big) + \frac{3}{2} \cdot \text{bd}(z')\Big) \tag{4.7}$$

For a tour $T^* \in$ OPT, define P_{T^*}, P'_{T^*} to be the corresponding r-peak(T^*) paths when viewing T^* as two paths between r and peak(T^*). Using (4.7), we write

$$\begin{aligned}
\sum_{z' \in Z} c(T_{\text{k-tsp}}(z')) &\leq (1+\varepsilon) \cdot \Big(\sum_{z' \in Z} \Big(\sum_{T^* \in \mathcal{T}^*_{z'}} c(T^*[\text{cell}(z')])\Big) + \frac{3}{2} \cdot \text{bd}(z')\Big) \\
&= (1+\varepsilon) \cdot \Big(\Big(\sum_{T^* \in \text{OPT}} \sum_{\substack{\text{cell}(z') \in \\ \mathcal{N}_\delta(z(T^*))}} c(T^*[\text{cell}(z')])\Big) + \sum_{z' \in Z} \frac{3}{2} \cdot \text{bd}(z')\Big) \\
&\leq (1+\varepsilon) \cdot \Big(\sum_{T^* \in \text{OPT}} \Big(\delta \cdot 2 \cdot c(r, \text{peak}(T^*)) + \text{regret}(P_{T^*}) + \text{regret}(P'_{T^*}) + 5 \cdot \varepsilon\Big) \\
&\quad + \sum_{z' \in Z} \frac{3}{2} \cdot \text{bd}(z')\Big) \\
&\leq (1+\varepsilon) \cdot \big(\delta \cdot \text{plb} + \text{regret(OPT)} + 6 \cdot \varepsilon \cdot \text{opt}\big) \\
&\leq \delta \cdot \text{plb} + \text{regret(OPT)} + 14 \cdot \varepsilon \cdot \text{opt},
\end{aligned}$$

where the second inequality follows from Claim 4.5, the third inequality holds by Lemma 2.8(iii), and the last inequality obtained by a very crude bound of $\delta \leq 1$, regret(OPT) $\leq$ opt, and $6 \cdot \varepsilon^2 \leq 6 \cdot \varepsilon$. ∎

Substituting the guarantee of the above claim in (4.6) finishes the proof of the lemma. ∎

4.3 Proof of Lemma 2.9. Putting together Lemma 4.3 and Lemma 4.4 gives us

$$\sum_{z \in Z} \sum_{j=1}^{t_z + t'_z} c(T_z^j) \leq \sum_{i=1}^{q} c(P'_i) + \sum_{z \in Z} c(T_{\text{peak}}(z)) + (\delta + 2 \cdot \rho) \cdot \text{plb} + 4 \cdot \varepsilon \cdot \text{opt} \tag{4.8}$$

By substituting the guarantee given by Lemma 4.6 for $\sum_{z \in Z} c(T_{\text{peak}}(z))$ in (4.8) we get

$$\sum_{z \in Z} \sum_{j=1}^{t_z + t'_z} c(T_z^j) \leq \sum_{i=1}^{q} c(P'_i) + \text{regret(OPT)} + (2 \cdot \delta + 2 \cdot \rho) \cdot \text{plb} + 19 \cdot \varepsilon \cdot \text{opt}. \tag{4.9}$$

Adding (4.9) to the guarantee given by Lemma 4.2 and recalling regret(OPT) $\leq \frac{\delta \cdot (1-\delta) \cdot \delta' \cdot \delta''}{2} \cdot$ opt (see Lemma 2.8(i)) finishes the proof of Lemma 2.9.

A Appendix. We prove the set systems defined in Section 3.3 are indeed matroids.

Given a graph $G = (V, E)$ with metric cost c and a subset $X \subseteq V$. Let $M = (E, I)$ be a set system on ground set E. A subset $F \subseteq E$ is independent if F is a forest and each component in F has at most one vertex from X. We prove M is a matroid which proves both M_1 and M_2 defined in Section 3.3 are matroids.

It is easy to see $\emptyset \in I$, and the hereditary property holds for M. Let us prove the exchange property. Let F_1 and F_2 be two independent sets, and $|F_1| < |F_2|$. We call the tree in both F_1 and F_2 that have a vertex in X, a ***special component***, otherwise we call the tree a ***non-special component***. Since $|F_1| < |F_2|$, the number of components in F_1 is strictly larger than the number of components in F_2. Also both F_1 and F_2 have the same number of special components. Therefore, there is an edge $e \in F_2 \setminus F_1$ that belongs to a cut edge of a non-special component A of F_1. Since $V(A) \cap X = \emptyset$, every tree in $F_1 \cup \{e\}$ has at most one vertex from X; hence, $F_1 \cup \{e\}$ is independent. This shows the exchange property holds for M.

References

[1] A. Adamaszek, A. Czumaj, and A. Lingas, *PTAS for k-Tour Cover Problem on the Plane for Moderately Large Values of k*, in International Symposium on Algorithms and Computation, Springer, 2009, pp. 994–1003.

[2] K. Altinkemer and B. Gavish, *Heuristics for unequal weight delivery problems with a fixed error guarantee*, Operations Research Letters, 6 (1987), pp. 149–158.

[3] S. Arora, *Polynomial time approximation schemes for euclidean traveling salesman and other geometric problems*, Journal of the ACM (JACM), 45 (1998), pp. 753–782.

[4] 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 Proceedings of the twenty-ninth annual ACM symposium on Theory of computing, 1997, pp. 275–283.

[5] 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, 2018, pp. 8:1–8:15.

[6] A. Becker, P. N. Klein, and A. Schild, *A ptas for bounded-capacity vehicle routing in planar graphs*, Berlin, Heidelberg, 2019, Springer-Verlag, p. 99–111.

[7] J. Blauth, V. Traub, and J. Vygen, *Improving the approximation ratio for capacitated vehicle routing*, Mathematical Programming, 197 (2023), pp. 451–497.

[8] A. Bompadre, M. Dror, and J. B. Orlin, *Probabilistic analysis of unit-demand vehicle routeing problems*, Journal of Applied Probability, 44 (2007), p. 259–278.

[9] N. Christofides, *Worst-case analysis of a new heuristic for the travelling salesman problem*, Operations Research Forum, 3 (1976).

[10] V. Cohen-Addad, A. Filtser, P. N. Klein, and H. Le, *On light spanners, low-treewidth embeddings and efficient traversing in minor-free graphs*, in 2020 IEEE 61st Annual Symposium on Foundations of Computer Science, 2020, pp. 589–600.

[11] G. B. Dantzig and J. H. Ramser, *The truck dispatching problem*, Management Science, 6 (1959), pp. 80–91.

[12] A. Das and C. Mathieu, *A quasipolynomial time approximation scheme for euclidean capacitated vehicle routing*, Algorithmica, 73 (2015), pp. 115–142.

[13] Z. Friggstad, R. Mousavi, M. Rahgoshay, and M. R. Salavatipour, *Improved approximations for a cvrp with unsplittable demands*, Mathematics of Operations Research, (2025).

[14] F. Grandoni, C. Mathieu, and H. Zhou, *Unsplittable euclidean capacitated vehicle routing: A $(2+\varepsilon)$-approximation algorithm*, in 14th Innovations in Theoretical Computer Science Conference., 2023, pp. pp–63.

[15] M. Haimovich and A. H. G. R. Kan, *Bounds and heuristics for capacitated routing problems*, Mathematics of Operations Research, 10 (1985), pp. 527–542.

[16] A. R. Karlin, N. Klein, and S. O. Gharan, *A deterministic better-than-3/2 approximation algorithm for metric TSP*, in Integer Programming and Combinatorial Optimization - 24th International Conference, IPCO 2023, 2023, pp. 261–274.

[17] A. R. Karlin, N. Klein, and S. O. Gharan, *A (slightly) improved approximation algorithm for metric TSP*, Oper. Res., 72 (2024), pp. 2543–2594.

[18] R. M. Karp, *Probabilistic analysis of partitioning algorithms for the traveling-salesman problem in the plane*, Mathematics of operations research, 2 (1977), pp. 209–224.

[19] C. Mathieu and H. Zhou, *Iterated tour partitioning for euclidean capacitated vehicle routing*, Random Structures & Algorithms, 62 (2022), pp. 1056 – 1075.

[20] C. Mathieu and H. Zhou, *A ptas for capacitated vehicle routing on trees*, ACM Trans. Algorithms, 19 (2023).

[21] J. S. Mitchell, *Guillotine subdivisions approximate polygonal subdivisions: A simple polynomial-time approximation scheme for geometric tsp, k-mst, and related problems*, SIAM Journal on computing, 28 (1999), pp. 1298–1309.

[22] Z. Nie and H. Zhou, *Euclidean capacitated vehicle routing in the random setting: A 1.55-approximation algorithm*, in 32nd Annual European Symposium on Algorithms (ESA 2024), vol. 308, 2024, pp. 91:1–91:15.

[23] J. Orestes Cerdeira, *Matroids and a forest cover problem*, Mathematical Programming, (1994), pp. 403–405.

[24] C. H. Papadimitriou, *The euclidean travelling salesman problem is np-complete*, Theoretical computer science, 4 (1977), pp. 237–244.

[25] S. Rathinam and R. Sengupta, *Matroid intersection and its application to a multiple depot, multiple tsp*, Institute of Transportation Studies, UC Berkeley, Institute of Transportation Studies, Research Reports, Working Papers, Proceedings, (2006).

[26] A. Schrijver, *Combinatorial Optimization: Polyhedra and Efficiency*, Springer, 2003.

[27] A. I. Serdyukov, *O nekotorykh ekstremal'nykh obkhodakh v grafakh*, Upravlyayemyye sistemy, 17 (1978), pp. 76–79.

[28] P. Toth and D. Vigo, *The vehicle routing problem*, Society for Industrial and Applied Mathematics, 2002.

[29] E. van Wijland and H. Zhou, *Faster approximation scheme for euclidean k-tsp*, in 40th International Symposium on Computational Geometry (SoCG 2024), 2024.