Article

The International Journal of Robotics Research
2020, Vol. 39(4) 389–401

DOI: 10.1177/0278364919893455
journals.sagepub.com/home/ijr
SAGE

Approximation algorithms for tours of orientation-varying view cones

Nikolaos Stefas, Patrick A Plonski and Volkan Isler

Abstract

This article considers the problem of finding a shortest tour to visit viewing sets of points on a plane. Each viewing set is represented as an inverted view cone with apex angle α and height h. The apex of each cone is restricted to lie on the ground plane. Its orientation angle (tilt) ϵ is the angle difference between the cone bisector and the ground plane normal. This is a novel variant of the 3D Traveling Salesman Problem with Neighborhoods (TSPN) called Cone-TSPN. One application of Cone-TSPN is to compute a trajectory to observe a given set of locations with a camera: for each location, we can generate a set of cones whose apex and orientation angles α and ϵ correspond to the camera's field of view and tilt. The height of each cone h corresponds to the desired resolution. Recently, Plonski and Isler presented an approximation algorithm for Cone-TSPN for the case where all cones have a uniform orientation angle of $\epsilon = 0$. We study a new variant of Cone-TSPN where we relax this constraint and allow the cones to have non-uniform orientations. We call this problem Tilted Cone-TSPN and present a polynomial-time approximation algorithm with ratio $O\left(\frac{1+\tan\alpha}{1-\tan\epsilon\tan\alpha}\left(1+\log\frac{\max(H)}{\min(H)}\right)\right)$, where H is the set of all cone heights. We demonstrate through simulations that our algorithm can be implemented in a practical way and that by exploiting the structure of the cones we can achieve shorter tours. Finally, we present experimental results from various agriculture applications that show the benefit of considering view angles for path planning.

Keywords

Path planning, view planning, approximation algorithms, geometric algorithms, euclidean traveling salesman problem, traveling salesman problem with neighborhoods

1. Introduction

Consider the task of an aerial vehicle charged with collecting images of a given set of locations. Such tasks arise in many applications such as crop monitoring, animal tracking, and road inspection. In this article, we study a novel coverage problem inspired by this scenario. We associate each measurement with an inverted cone apexed at the location of interest. The height of the cone is associated with the desired resolution and the apex angle corresponds to the camera's field of view (FOV). In other words, each cone encodes the set of view points from which a target can be imaged at a desired location. See Figure 1. The task is to visit a given set of cones so as to ensure that all locations are covered.

Coverage is a fundamental problem in robotics and has been studied extensively (Choset, 2001; Galceran and Carreras, 2013). A coverage tree structure was proposed in Sadat et al. (2014) allowing for online, non-uniform adaptive coverage with an unmanned aerial vehicle (UAV). However, this strategy cannot provide any guarantees on the trajectory length which coupled with the UAV's limited battery might result in an incomplete tour. The work of Cheng et al. (2008) provides a constant factor solution for a UAV covering an urban building with a camera sensor. In contrast to our work, the orientation angle of the camera was fixed to look downwards. Complete coverage of a tree surface using right angular cones was used by Stefas et al. (2016) for a UAV flying at a low altitude inside an orchard. However, non-uniform tilt angle views were not considered. A sampling-based methodology to generate trajectories for coverage of 3D objects with unmanned underwater vehicles was presented by Englot and Hover (2013). Similar to our work, the authors achieved short trajectories and applied their method to underwater ship hull inspection. More recently, informative path planning problems in which mobile robots aim to maximize an objective function

University of Minnesota, Minneapolis, MN, USA

Corresponding author:
Nikolaos Stefas, Department of Computer Science and Engineering, University of Minnesota, 200 Union Street SE, Minneapolis, MN 55455, USA.
Email: stefa125@umn.edu

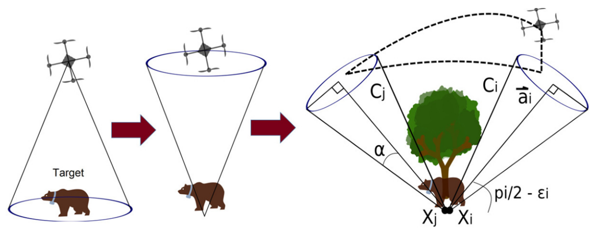

Fig. 1. In order for a UAV to capture images of a target animal it needs to enter an inverted view cone C_i, positioned at X_i with orientation vector $\vec{a}_i$ and apex angle $\alpha \leqslant \frac{camera_{FOV}}{2}$. The height h_i of the cone is relative to the desired resolution. If visibility is occluded, the view cone C_i may need to be tilted by an angle ϵ_i.

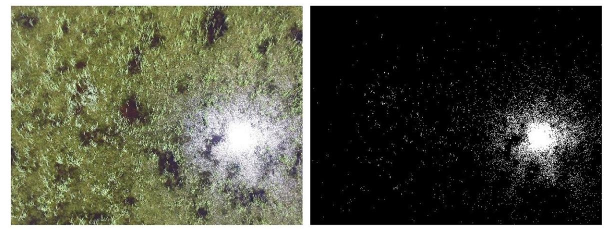

Fig. 2. When covering watery fields such as wild rice (left image), sunlight reflection can be a problem (right image). By choosing our view angles carefully we can reduce sunlight specularities.

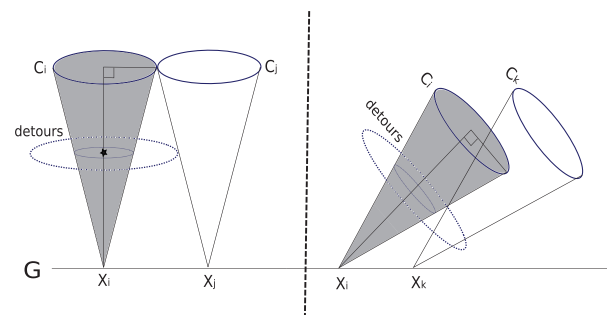

Fig. 3. When using the non-Tilted Cone-TSPN strategy we plan two detours, an inner and outer concentric circular path to ensure all cones that intersect cone C_i are visited. This is illustrated with cones C_i and C_j (left). However, if cone C_i tilts, these two detours can no longer guarantee that all cones intersecting it will be visited because the relative arrangement of the cones can change. This is illustrated with cones C_i and C_k (right). Cone C_k can be placed such that it intersects cone C_i but lies in between the two previously planned detours and is not covered by them.

related to information gain were studied by Singh et al. (2009) and Low et al. (2008). When the locations to be covered are known ahead of time, we can compute an optimal trajectory to visit them.

The basic task of visiting a given set of locations is the Traveling Salesman Problem (TSP) which admits a polynomial-time approximation scheme (PTAS) for the Euclidean version (Arora, 1998; Mitchell, 1999). If we incorporate sensor footprint as a set of given areas instead of points we wish to visit, then we have the Traveling Salesman Problem with Neighborhoods (TSPN), which is APX-hard (de Berg et al., 2005; Safra and Schwartz, 2006). TSPN in two dimensions is a well-studied problem and many researchers have provided approximation algorithms for various cases (Reinelt, 1994). Euclidean TSPN has been shown to admit a PTAS if the neighborhoods have a well-defined structure (Dumitrescu and Mitchell, 2001; Mitchell, 2007). A constant factor approximation was shown in Dumitrescu and Tóth (2015) for arbitrary, planar disks. TSPN in three dimensions is a more difficult problem. A PTAS was provided by Bodlaender et al. (2009) for disjoint polygons of comparable size and a quasipolynomial-time approximation scheme (QPTAS) was provided by Chan and Elbassioni (2011) for α-fat, weakly disjoint neighborhoods. Dumitrescu and Tóth (2016) presented a constant factor approximation when the neighborhoods take the form of unit balls, lines, or planes. Recently, in Plonski and Isler (2019), we presented a polynomial-time approximation algorithm for the 3D TSPN with intersecting neighborhoods when the neighborhoods take the form of right angular (non-tilted) cones and their apex points lie on a planar surface. We called this problem Cone-TSPN. Our approach was based on the idea that by intersecting the cones with a set of horizontal planes we can reduce the problem to that of equal size disks on the plane with a number of upper bounded detours. There are instances when it is desirable to have tilted cones. For example, carefully chosen tilted views might be required if visibility of the target area is limited (see Figure 1) or when covering reflective surfaces (see Figure 2). In Section 2, we demonstrate with quantitative results that in such cases we can obtain more visual information (i.e., better views) if we introduce cones with different tilt angles. We address this problem in this article.

The Cone-TSPN strategy presented in Plonski and Isler (2019) is not directly applicable for input view cones of varying orientation (or tilt) and cannot provide theoretical guarantees. Removing the assumption that the cones are right angular (not tilted) requires addressing a number of challenges. When the cones are allowed to tilt, their intersection with a horizontal plane is no longer a disk but an ellipse. This means that we can no longer use a PTAS TSPN tour for disks to visit the cones on the plane. Instead, we use a PTAS TSP on their center and bound the length of this tour with respect to the optimal solution. More importantly, the relative arrangement of the cones change as they tilt and the previously planned detours cannot guarantee complete coverage of all cones (see Figure 3 for an example). We need to modify and possibly add additional detours (see Section 6 for details). These challenges arise when the cones have uniform orientation angles. The problem becomes even more challenging if we let the cones have non-uniform orientation angles. We address this challenge by grouping the cones by similar angles and further modifying the detour strategy to guarantee coverage of all cones in the same group (see Section 7). In this work, we extend Cone-TSPN and provide a polynomial-time approximation algorithm for tilted input view cones.

The rest of the article is organized in the following way. The problem statement is presented in Section 3. Our analysis starts with disjoint cones of similar height and identical (uniform) orientation in Section 4. Then, we study the cases where the cones do not have similar heights in Section 5 and are not disjoint in Section 6. We conclude our analysis with non-disjoint cones with varying (non-uniform) orientation in Section 7 and evaluate the performance of our algorithm in Section 8. Complete proofs for the analyses of Sections 4, 5, 6, and 7 are included in the appendix. Finally, in Section 2, we show our quantitative results for two different agriculture applications with different view angles and discuss future work in Section 9.

We conclude this section with a summary of our contributions.

1. We present a polynomial-time approximation algorithm that solves the Cone-TSPN problem for input view cones of varying orientation (or tilt).
2. We provide an implementation and a detailed analysis of its performance.
3. We demonstrate that tilted cone views are useful in different agriculture applications for which top-down views (non-tilted cones) may not be sufficient.

2. Motivating field applications

Currently, most applications that perform visual coverage of an area with a UAV are restricted to right angle (non-tilted), uniform, top-down views. However, this can result in significant loss of information. To demonstrate this we consider two different scenarios. First, we consider the problem of visual coverage of reflective surfaces. We demonstrate that by choosing our angle views carefully we can reduce sunlight specularities. Second, we consider the problem of visual inspection of a target in difficult to see areas. We show a scenario where tilted view angles can successfully obtain visual information of a target object or area even if it lies below another object blocking the top-down views (e.g. bridges, trees).

2.1. Reflective surface coverage

We performed a set of experiments that verify our claim that coverage of reflective areas requires tilted view cones to avoid direct sunlight reflection. We covered a 30×30 m^2 area over a lake at an altitude of 10 m with three different view angles. We will refer to the view angle with the least amount of sunlight reflection in the camera as the best view angle. Similarly, the worst view angle is that with the greatest amount of sunlight reflection. Using the 0°, non-tilted views as baseline (top-down views), we compared the amount of sunlight in the images using pixel intensities for over 1,000 images (see Figure 4).

Fig. 4. Reducing sunlight specularities in images using tilted view cones. We covered a lake area with a UAV taking images (left column) and used pixel intensities to quantify the amount of sunlight on the images (right column). The top-down views were used as a baseline (top row). The worst view angles had 671.75% more sunlight than the top-down views (middle row). The best view angles had 81.57% less sunlight than the top-down views (bottom row).

At the time of the experiments, the Sun polar angle was 35° and the Sun azimuthal angle was 250° (from the magnetic north). By using the law of reflection, we calculated the Sun angles and identified that the best view angle was the one with a polar angle of 55° and azimuthal angle of 70°. The worst view angle had a polar angle of 55° and azimuthal angle of 250°. By covering the area with the best view angles, the average amount of sunlight specularities in the images was reduced by 81.57%. However, coverage with the worst view angles increased the average amount of sunlight specularities in the images by 671.75%. These results validate our claim that we can reduce sunlight specularities when covering reflective surfaces by choosing our coverage views carefully.

2.2. Visual inspection of a target

We performed a set of experiments to verify our claim that tilted view cones can provide more visual information of a target object if its occluded by other objects and visibility is limited. During this set of experiments we chose a red ball as our target object under a tree cluster that blocks visibility. The tree cluster could fit on a cylinder of about 20 m in diameter and 25 m in height. We generated two coverage plans. The first was top-down view coverage. The second was tilted view coverage. The center of the cluster on the ground

Fig. 5. GPS trajectories of the two coverage plans. Top-down coverage was performed with a double grid (blue) with 50 m edges and at 25 m altitude. Tilted coverage was performed with a circular pattern at radius 12.5 m, altitude of 12.5 m, and camera tilt of 60°.

was chosen as the reference point for both coverage plans. In order to quantify how much visual information we obtain, we placed a red ball of 1 m diameter at the center of the cluster and counted the number of views that detected it.

Top-down views were acquired with a square double grid coverage pattern. The altitude chosen was 25 m to avoid hitting the tree (see Figure 5). Our camera had a 90° FOV and at an altitude of 25 m the square grids edge was about 50 m. Pixel resolution was $3,000 \times 3,000$ and, thus, we cover 1 m^2 with 60^2 pixels (the ball area). We chose an overlap of 80% and for each 50 m line we took 10 images every for a total of 120 images. For every image acquired we performed simple color segmentation to detect the red ball and counted the number of pixels. If we counted more than 1,500 pixels (equivalent to seeing over 41% of the ball) in a single image, we successfully detected the red ball. Out of all the images acquired with the top-down view coverage plan the red ball was present in only 1 (see Figure 6). It is worth noting that in both views the ball was not fully visible.

Tilted views were acquired uniformly on a circle around the tree with a tilt of 60°. For a camera tilt of 60° and a distance of 25 m (with respect to the red ball) altitude was chosen to be 12.5 m. Similarly, the radius of the circle pattern was 12.5 m. We chose an 87.5% overlap (with respect to the bounding circle) and acquired 32 views uniformly. Using the same procedure as before, we detected the red ball in five images acquired from the tilted view coverage plan (see Figure 6). The ball was almost fully visible in two of the images. These results validate our claim that we can obtain more visual information in certain scenarios when using tilted side views.

3. Problem statement

Our problem, Tilted Cone-TSPN, can be formulated in the following way. We are given $C = (C_1, \ldots, C_n)$, $\vec{A} = (\vec{a}_1, \ldots, \vec{a}_n)$, $X = (X_1, \ldots, X_n)$, $H = (h_1, \ldots, h_n)$ where C is a set of cones $C_1, \ldots, C_n$ with fixed apex angle α. Cone C_i has apex point X_i on the ground plane G with normal $\vec{n}$ and orientation vector $\vec{a}_i$ of length h_i such that $\vec{n}\angle\vec{a}_i = \epsilon_i$. The goal is to compute a minimum length trajectory T which intersects all cones in C (see Figure 7).

We present a strategy called Orientation-Visit (Algorithm 5), which has an approximation factor

$$O\left(\frac{1+\tan\alpha}{1-\tan\epsilon\tan\alpha}\left(1+\log\frac{\max(H)}{\min(H)}\right)\right) \tag{1}$$

with $|\epsilon| + |\alpha| < \frac{\pi}{2}$, which means that the cones (and by extension the UAV) do not touch the ground.

The tangent terms in Equation (1) behave well in practical situations. A camera has usually around $\frac{\pi}{2}$ FOV, which translates to $\alpha \leqslant \frac{\pi}{4}$ and $\tan\alpha \leqslant 1$. If we set $\epsilon = \alpha - \frac{\pi}{30}$, which means that the view cones will almost touch the ground, we have $\tan\epsilon \leqslant 0.9$ and, thus, $\frac{1+\tan\alpha}{1-\tan\epsilon\tan\alpha} \leqslant 20$. Note that if we have non-uniform orientations $\epsilon = \arg\max_{a_i} \vec{n}\angle\vec{a}_i$.

4. Disjoint cones of similar height and identical orientation

In this section, we present the strategy for the case where the cones are disjoint, have similar heights, and identical orientation. We first present a method to obtain upper and lower bounds of the swept area of a tour using conic volumes. These bounds help us remove dependence on the number of cones when computing the performance of our strategy. The strategy is outlined in Algorithm 1. `Tilted Slice-Visit` solves Tilted Cone-TSPN for disjoint cones of similar height and identical orientation angle ϵ by fixing a coverage plane P_{h_t} at height h_t and visiting the cone bisectors on that plane with a TSP tour. Finally, if the optimal tour T^* achieves maximum height h^*, we assume that there exists an estimate height $\hat{h}$ in the direction of the cone orientation vector $\hat{a}$ such that $h^* \leqslant \hat{h} \leqslant 2h^*$ (its existence is proven in Plonski and Isler (2019)).

Lemma 1. *Let the optimal tour T^* have length L^* and maximum height h^*. Let its projection onto plane G with normal $\vec{n}$ be T^*_G. For maximum cone height h_{max} and an estimate height $\hat{h}$ such that $\hat{h} \geqslant h_{max}$, $\hat{h} \leqslant 2h^*$, the Minkowski sum sweep volume $f(T^*_G, \hat{h})$ of a cone C with apex and orientation angles α and ϵ such that $|\epsilon| + \alpha \in (0, \frac{\pi}{2})$ traveling along T is upper bounded by $f(T^*_G, \hat{h}) \leqslant L^*\hat{h}^2\tan\alpha\left(1 + \frac{5\pi}{6}\tan\alpha\right)$.*

Proof. The volume swept by a cone C with apex angle α, orientation angle ϵ, and height h along a path of length $L^*_G \in G$ can be split into three parts (see Figure 8). First,

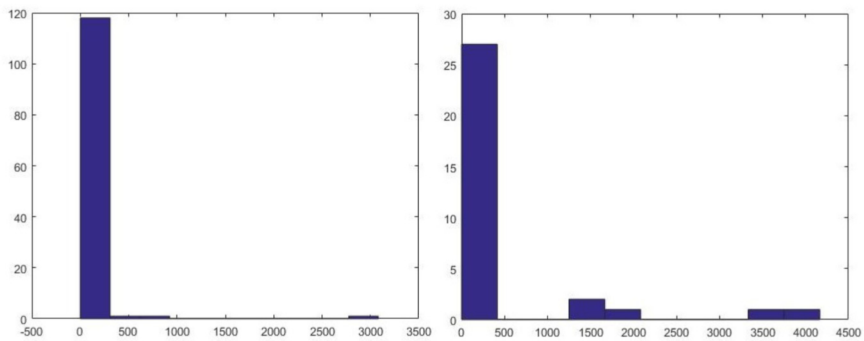

Fig. 6. Number of red ball pixels detected for top-down coverage views (left) and tilted coverage views (right).

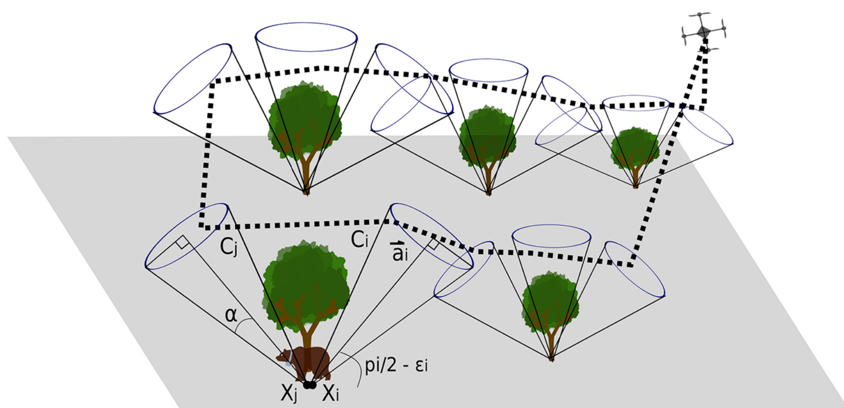

Fig. 7. Given a set of inverted view cones C with apex angle α, heights H, tilts E, orientation vectors $\vec{A}$, and apex points X located at points of interest the goal is to find the shortest tour to visit them.

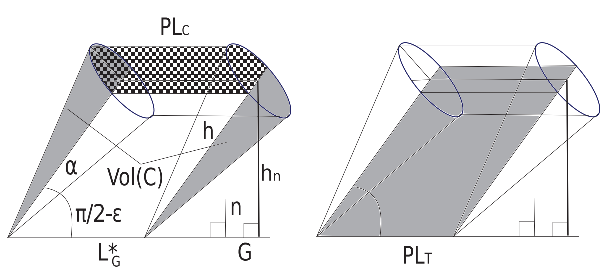

Fig. 8. The sweep volume of a cone C along a path of length L_G^* can be split into three parts. The checkered pattern area of a semicircle swept by the upper half base of the cone. The triangle area PL_T is swept by the largest inscribing triangle on cone C. The gray area $Vol(C)$ includes the points covered by two half cones located at the start and end of L_G^*.

we have $vol(C)$ as two halves of the volume of a cone covering the points at the starting and ending locations of L_G^* not covered by any other cone along L^*. Second, we have PL_C, the area of a semicircle along the path L_G^*. Here PL_C is the area swept by the upper half base of the cone in the direction of the normal of G. If L_G^* is a straight line, its volume looks like that of a slanted half cylinder. Third, we have PL_T, the area of a triangle along L_G^*. Here PL_T is the area swept by the largest inscribing triangle on cone C. Finally, we note that $\hat{h} \geqslant h_n = \vec{a} \cdot \vec{n}$. □

Lemma 2. *Let the optimal tour T^* have length L^* and maximum height h^*. Let its projection onto plane G with normal $\vec{n}$ be T_G^*. For maximum cone height h_{max} and an estimate height $\hat{h}$ such that $\hat{h} \geqslant h_{max}$, $\hat{h} \leqslant 2h^*$, there exists a constant C_v such that the Minkowski sum sweep volume $f(T_G^*, \hat{h})$ of a cone C with apex and orientation angles α and $\epsilon = 0$ such that $|\epsilon| + \alpha \in (0, \frac{\pi}{2})$ traveling along T is lower bounded by $f(T_G^*, \hat{h}) \geqslant C_v \tan^2 \alpha \sum_{h_i \in H} h_i^3$.*

Proof. The volume of a cone swept out by a tour visit can be represented as the intersection between two cones $C_{i'}$, $C_{j'}$ of equal height h_i and apex angle α, with their apex points at distance $h_i \tan \alpha$. The volume of this intersection is proportional to the volume of $C_{i'}$, say by a constant number C_v. For a single cone visit we have $\frac{\pi r_i^2 h_i}{3} = \frac{\pi h_i^3 \tan^2 \alpha}{3}$ and the tour visits all cones with heights $h_i \in H$ (see Figure 9). □

Now that we have obtained the bounds for the swept area of a tour, we study how they change as the cones tilt. The following two lemmas show how the cone orientation angle ϵ affects the lower bound. The upper bound remains the same.

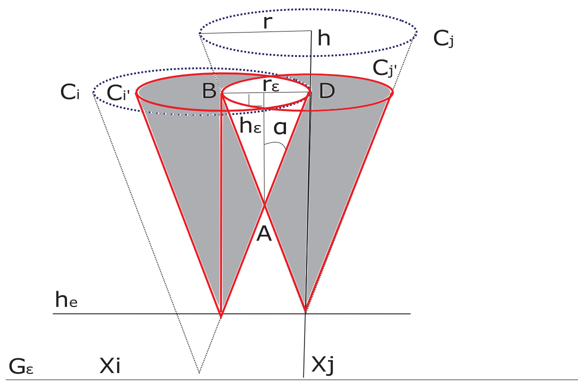

Fig. 9. The volume of intersection between two cones $C_{i'}$, $C_{j'}$ of equal height is proportional to the volume of $C_{i'}$ by a constant number C_v.

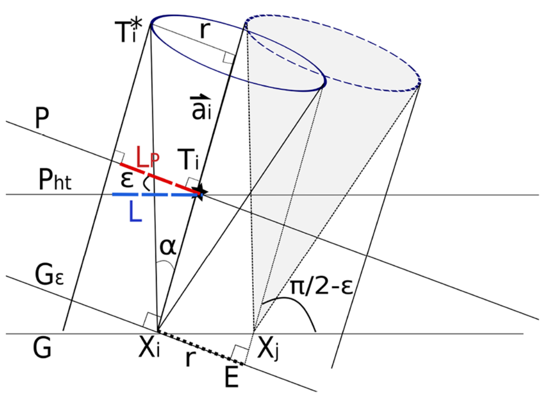

Fig. 10. Tilted Slice-Visit. TSP will visit the cone at T_i, with $|X_iT_i| = h_t$. Here T_i is not further away than $h_i \tan\alpha$ from T_i^* projected along $\vec{a}$ onto P_{h_t}.

Lemma 3. *Let cones C_i, C_j have apex and orientation angles α and ϵ, height h_i, cap radius $r = h_i \tan\alpha$, and apex points $X_i, X_j \in$ plane G with normal $\vec{n}$. Let G_ϵ be a 2D plane with normal $\vec{n}_\epsilon$ passing through X_i such that $\vec{n}\angle\vec{n}_\epsilon = \epsilon$ and $\vec{a}_i\angle\vec{n}_\epsilon = 0$. The relative arrangement of two cones tilted by angle ϵ with $\vec{a}_j$ intersecting the cap of c_i is identical to that of two non-tilted cones with their apex points $X_i, X_j \in G_\epsilon$ and one of them elevated from G_ϵ by $h_e = r\tan\epsilon$.*

Proof. In Figure 10, ΔX_iEX_j is a right triangle with $\hat{E} = \frac{\pi}{2}$: $\tan\epsilon = \frac{|EX_j|}{|X_iE|} = \frac{|EX_j|}{r} \Leftarrow |EX_j| = r\tan\epsilon$. □

Lemma 4. *Let the optimal tour T^* have length L^* and maximum height h^*. Let its projection on plane G with normal $\vec{n}$ be T_G^*. For maximum cone height h_{max} and an estimate height $\hat{h}$ such that $\hat{h} \geqslant h_{max}$, $\hat{h} \leqslant 2h^*$, there exists a constant C_ϵ such that the Minkowski sum sweep volume $f(T_G^*, \hat{h})$ of a cone C with apex and orientation angles α and ϵ such that $|\epsilon| + \alpha \in (0, \frac{\pi}{2})$ traveling along T is lower bounded by $f(T_G^*, \hat{h}) \geqslant C_\epsilon \tan^2\alpha \sum_{h_i \in H} h_i^3$, where $C_\epsilon = C_v(1 - \tan\alpha \cdot \tan\epsilon)^3$.*

Algorithm 1. Tilted Slice-Visit

Input: x_0, C, H, $\hat{A}$, X, ϵ, α
Output: Tilted Cone-TSPN tour T

1: Define a plane P_{h_t} that is parallel to G and elevated by coverage height $h_t = h_{min}$ in the direction of $\vec{a}$
2: Intersect all orientation vectors with $P_{\hat{h}}$, the result is a number of points on P_{h_t}
3: Approximate TSP tour T_{h_t} that visits all points on P_{h_t} with starting point x_0 (using, e.g., a PTAS for Euclidean points)
4: Connect P_{h_t} with G using a vertical double line segment at x_0

Proof. Given two tilted by ϵ cones C_i, C_j, by Lemma 3 this is equivalent to two non-tilted cones where C_j is elevated by $h_e = r\tan\epsilon$. Create a new cone $C_{j'}$ with apex point X_j, apex angle α, height $h_i - h_e = h(1 - \tan\alpha\ \tan\epsilon)$, and radius $r_\epsilon = h\tan\alpha(1 - \tan\alpha\tan\epsilon)$ (see Figure 9). Create a new cone $C_{i'}$ with apex point $X_{i'}$ lying at the intersection between h_e and $|X_iD|$, with radius r_ϵ, apex angle α and height $h_i - h_e$. The volume of intersection between C_i, C_j is larger than the volume of intersection between $C_{i'}$, $C_{j'}$. Applying Lemma 2 on $C_{i'}$, $C_{j'}$ and noting that $C_\epsilon = C_v(1 - \tan\alpha\tan\epsilon)^3$ produces the desired lower bound. □

Now that we have obtained the bounds for the swept area of a tour relative to the orientation angle, we can compute the performance of our strategy in Algorithm 1.

Lemma 5. *Let an input set C of n disjoint cones have orientation and apex angles ϵ and α, heights H, and coverage height h_t. If Tilted Cone-TSPN tour T is computed with algorithm Tilted Slice-Visit using an $(1+\beta)$ approximation, then $\frac{L}{1+\beta} \leqslant 2h_t + \frac{L^* + 2n\tan\alpha}{\cos\epsilon}\mathrm{mean}(H)$.*

Proof. Let the optimal tour T^* with length L^* visit cone C at point T_i^*, lying at cone height h_i. Here T_i^* cannot be further than $r = h_i\tan\alpha$ from the orientation vector $\vec{a}$ (see Figure 10). Our strategy T with length L visits the cone at point T_i, which is a point on the orientation vector $\vec{a}$. Define plane P such that its normal is parallel to $\vec{a}$ and it passes through X_i. Project T_i^* along $\vec{a}$ onto P and call the resulting point $T_{i,P}^*$. On P $|T_{i,P}^*T_i| \leqslant h_i\tan\alpha$, then

$$L_P - L_P^* \leqslant \sum_{\forall c_i \in C} h_i \tan\alpha \qquad (2)$$

The projection of a tour with length L^* onto P cannot make it longer than

$$L_P^* \leqslant L^* \qquad (3)$$

Furthermore, the projection of tour on P with length L_P onto P_{h_t} such that $P\angle P_{h_t} = \epsilon$ cannot get longer than a factor of $\cos\epsilon$:

$$L \leqslant \frac{L_P}{\cos\epsilon} \qquad (4)$$

Combining Equations (2), (3), and (4), we obtain

$$L \leqslant \frac{L^* + \sum_{\forall c_i \in C} h_i \tan\alpha}{\cos\epsilon} \tag{5}$$

Connecting $x_0 \in G$ to P_{h_t} requires two additional line segments of length at most $2h_t$. Finally, note that $\sum_{h_i \in H} h_i = n\text{mean}(H)$. ☐

Theorem 6. *Let an input set C of n disjoint cones have orientation and apex angles ϵ and α and heights H. For an estimate height $\hat{h} \geqslant h_{max}$, $h^* \leqslant \hat{h} \leqslant 2h^*$ and $|\epsilon| + \alpha \in (0, \frac{\pi}{2})$, if the strategy Tilted Slice-Visit solves Tilted Cone-TSPN using a $(1+\beta)$ approximation, then it has an approximation factor*

$$(1+\beta)\left(\frac{1}{\cos\epsilon}\left(\frac{\hat{h}}{\text{mean}(H)}\right)^2\left(1+\frac{5\pi}{6}\tan\alpha\right)\right)$$

Proof. From Lemmas 1 and 4 and noting that $\sum_{h_i \in H} h_i^3 \geqslant n \cdot \text{mean}(H)^3$, we have

$$n\tan\alpha \leqslant \frac{L^*\hat{h}^2(1+\frac{5\pi}{6}\tan\alpha)}{C_\epsilon \text{mean}(H)^3} \tag{6}$$

Substituting with Lemma 5, we obtain

$$\frac{L}{1+\beta} \leqslant 2h_t + \frac{L^*}{\cos\epsilon} + 2\frac{\text{mean}(H)}{\cos\epsilon}\frac{L^*\hat{h}^2\left(1+\frac{5\pi}{6}\tan\alpha\right)}{C_\epsilon \text{mean}(H)^3} \tag{7}$$

Noting that $h_t \leqslant \hat{h}$, $h_t \leqslant 2h^* \leqslant L^*$ gives the resulting bounds.

5. Disjoint cones of identical orientation

In this section, we present the strategy for the case where the cones are disjoint, have different heights, and identical orientation. If the cones have different heights, then Algorithm 1 may perform poorly owing to being restricted to a coverage height of h_{min}. This is addressed by splitting the cones into a number of height bins such that the requirements of Lemma 1 are met and perform Tilted Slice-Visit on each. The strategy is outlined in Algorithm 2.

Theorem 7. *Let an input set C of n disjoint cones have orientation and apex angles ϵ and α and heights H. For an estimate height $\hat{h} \geqslant h_{max}$, $h^* \leqslant \hat{h} \leqslant 2h^*$ and $|\epsilon| + \alpha \in (0, \frac{\pi}{2})$, if the strategy Tilted Height-Visit solves Tilted Cone-TSPN using a $(1+\beta)$ approximation, then it has an approximation factor*

$$(1+\beta)\left(\frac{1}{\cos\epsilon}\left(1+\frac{5\pi\tan\alpha}{6}\right)\left\lfloor 1+\log_2\frac{\hat{h}}{h_{min}}\right\rfloor\right)$$

Proof. Given coverage height h_t, and the approximation factor from Theorem 6, we have $\left\lfloor 1+\log_2\frac{\hat{h}}{h_{min}}\right\rfloor$ height bins

Algorithm 2. Tilted Height-Visit

Input: x_0, C, H, ϵ, α
Output: tiltedCone-TSPN tour T
$i = 0$
repeat
  Create bin with height range $B_i = [2^i h_{min}, 2^{i+1} h_{min})$
  For all cones with height $\in B_i$ call Algorithm 1 and find tour T_{B_i} with starting point x_0
  $i = i + 1$
until $2^i h_{min} > h_{max}$
Connect all tours T_{B_i} with a vertical line segment at x_0

Algorithm 3. Tilted Height-Select

Input: C, H, ϵ, α
Output: Tilted Cone-TSPN tour T
Sort cones into a set C_{sort} from shortest to tallest based on cone height h
$MIS = \emptyset$
repeat
  Select the first cone $C_1 \in C_{sort}$, this is the shortest cone in the set
  $MIS = MIS \cup C_1$
  Remove from C_{sort} cone C_1 and all cones intersecting with it
until $C_{sort} = \emptyset$
Call Algorithm 4 with input MIS

and we compute a subtour for each bin. Finally, note that $\hat{h} \leqslant 2h_t$ and $\text{mean}(H) \leqslant \frac{3}{2}h_t$. ☐

6. Non-disjoint cones of identical orientation

In this section, we present the strategy for the case where the cones are not disjoint, have different heights, and identical orientation. As we already have a method for the disjoint case, we can handle intersections of cones with identical orientation simply by selecting their maximal independent set (MIS) (Dumitrescu and Mitchell, 2001; Elbassioni et al., 2006). The cones in the MIS intersect all cones $\in C$. The strategy is outlined in Algorithm 3, which is an extension of Algorithm 2 for the case where the cones are not disjoint. `Tilted Height-Select` constructs the MIS by selecting cones greedily based on height. Then it computes a tour to visit the cones in the MIS and adds the necessary detours to ensure all cones are visited.

Lemma 8. *Given an input set C with orientation and apex angles ϵ and α let the MIS be a subset of C such that the cones $\in$ MIS do not intersect one another and collectively intersect all cones $\in$ C. If the MIS is selected using Tilted Height-Select, then Algorithm 6 adds $k = \lceil \frac{2}{1-\tan\alpha\tan\epsilon} \rceil$ detours of length $\leqslant (8k\pi + 4)h_t\tan\alpha$ that visit every cone.*

Proof. At the coverage height h_t, the maximum distance between the orientation vectors $\vec{a}_i$, $\vec{a}_j$ of a cone $c_i \in MIS$ and a cone intersecting it $c_j \notin MIS$ is $4h_t\tan\alpha$, because

Algorithm 4. Tilted Height-Visit Intersect

Input: $C, H, x_0, \epsilon, \alpha, \vec{A}$
Output: Tilted Cone-TSPN tour T
Truncate all cones to not be taller than $\hat{h} = h_{max}$
$i = 0$
repeat
  Create bin with height range $B_i = [2^i h_{min}, 2^{i+1} h_{min})$
    For all cones with height $\in B_i$ call Algorithm 1 and find tour T_{B_i} with starting point x_0.
  for each cone c_i visited in T_{B_i} **do**
    Add $k = \lceil \frac{2}{1 - \tan\alpha \tan\epsilon} \rceil$ circumference detours perpendicular to $\vec{a}$ such that detour $j \in [1, k]$ is centered at T_i and has radius $j2(h_t - h_e)\tan\alpha$
  end for
  $i = i + 1$
until $2^i h_{min} > h_{max}$
Connect all tours T_{B_i} with a vertical line segment at x_0

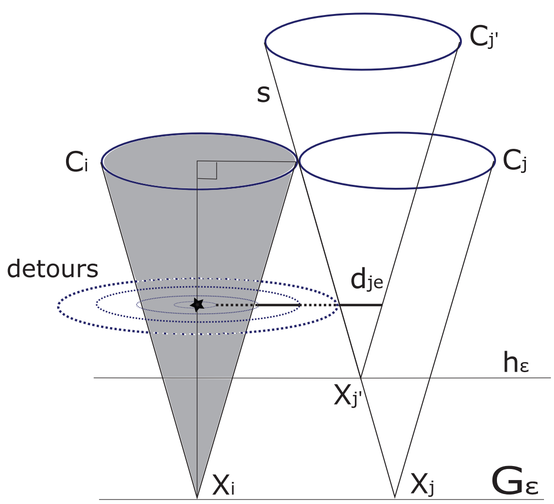

Fig. 11. The furthest cone C_j can be from C_i while still intersecting it results in have a single intersection point lying along the straight line s originating from its apex X_j.

$2h_t \leqslant h_i$. If the cones are tilted by ϵ, then from Lemma 3 the relative arrangement of two such cones is identical to that of two non-tilted cones with one of them $\notin MIS$, say $c_j{}'$, elevated from the plane by $h_e = h_i \tan\alpha \tan\epsilon$. The maximum distance between the orientation vectors $\vec{a}_i$, $\vec{a}_{j'}$ is $4h_t \tan\alpha$. At height h_t, cone $c_{j'}$ has diameter $d_{je} = 2(h_t - h_e)\tan\alpha$, thus we can guarantee coverage from $\vec{a}_i$ by adding circumference paths at a right angle, originating at T_i and having radii that are increments of d_{je}. The total length is less than $k8\pi h_t \tan\alpha + 4h_t \tan\alpha$, where k is a constant integer such that $k = \left\lceil \frac{4h_t \tan\alpha}{2(h_t - h_e)\tan\alpha} \right\rceil = \lceil \frac{2}{1 - \tan\alpha \tan\epsilon} \rceil$ (see Figure 11). □

Now that we have computed the cost of the additional detours we can calculate the performance of Algorithm 3.

Theorem 9. Let an input set C of n disjoint cones have orientation and apex angles ϵ and α and heights H. For an estimate height $\hat{h} \geqslant h_{max}$, $h^* \leqslant \hat{h} \leqslant 2h^*$ and $|\epsilon| + \alpha \in (0, \frac{\pi}{2})$, if the strategy Tilted Height-Select solves Tilted Cone-TSPN using a $(1 + \beta)$ approximation, then it has an approximation factor

$$(1+\beta)\left(\frac{18k}{\cos\epsilon}\left(1 + \frac{5\pi \tan\alpha}{6}\right)\left\lfloor 1 + \log_2 \frac{\hat{h}}{h_{min}} \right\rfloor\right)$$

where $k = \lceil \frac{2}{1 - \tan\alpha \tan\epsilon} \rceil$.

Proof. Similar to Theorem 6 we can add the new detours to the length of the detour in Lemma 5:

$$\frac{L}{1+\beta} \leqslant 2h_t + \frac{L^*}{\cos\epsilon} + 2\frac{\text{mean}(H) + h_t k(8\pi + 4)}{\cos\epsilon} \frac{L^* \hat{h}^2 \left(1 + \frac{5\pi}{6}\tan\alpha\right)}{C_\epsilon \text{mean}(H)^3}$$

Similar to Theorem 7, we have $\left\lfloor 1 + \log_2 \frac{\hat{h}}{h_{min}} \right\rfloor$ such tours. □

7. Non-disjoint cones of varying orientation

In this section, we present the strategy for the case where the cones are disjoint, have different heights, and different orientation. In order to handle the case where the cones have different orientation angles, we split them into different orientation sets. We create a number of orientation sets such that all cones have orientation angle difference θ such that $\epsilon_i \angle \epsilon_j = \theta \leqslant \frac{\alpha}{2}, \forall i, j,\ \theta + \alpha \in (0, \frac{\pi}{2})$. For each such set, we show that we only need to add one additional circumference detour to our previous strategy. The strategy is outlined in Algorithm 5. Note that we now have different orientation angles so $\epsilon = \arg\max_{a_i} \vec{n} \angle \vec{a}_i$.

Lemma 10. *Let two intersecting cones $C_i \in MIS$, $C_j \notin MIS$ have apex angle α, heights h_i, h_j and orientation angles ϵ_i and ϵ_j, with $\theta = \epsilon_i \angle \epsilon_j \leqslant \frac{\alpha}{2}$. If coverage height $h_t \geqslant h_i \frac{\sin\theta}{\sin(2\alpha + \theta)} + r\tan\epsilon$, we only need to add one additional circumference detour to the strategy outlined in Algorithm 3 centered at the center of the cone at height h_t, at a right angle with respect to $\vec{a}_i$ and at distance $4h_t \tan\alpha + 2(h_t - r\tan\epsilon)tan\alpha$ to guarantee coverage of all cones intersecting those $\in MIS$ (see Algorithm 5 and Figure 12).*

Proof. First we note that if the cones are tilted by ϵ, then from Lemma 3 the relative arrangement of two such cones is identical to that of two non-tilted cones with one of them $\notin MIS$, say $c_j{}'$, elevated from the plane by h_e. Thus, without loss of generality we can assume that on the relative arrangement between cones C_i and C_j, $\epsilon_i = \frac{\pi}{2}$. In Figure 12, applying law of sines on $\Delta I X_k X_j$ gives $|X_k X_j| = |IX_k| \frac{\sin\theta}{\cos(\alpha + \theta)} = \frac{r_i}{\sin\alpha} \frac{\sin\theta}{\cos(\alpha + \theta)}$. Applying the law of sines on $\Delta X_k M X_j$, gives $|X_k M| = \frac{r_i}{\sin(2\alpha + \theta)} \frac{\sin\theta}{\sin\alpha}$. Triangle $\Delta X_k HM$ is a right-angled triangle, thus $|HX_k| = hi \frac{\sin\theta}{\sin(2\alpha + \theta)}$, where $r_i = h_i \tan\alpha$. Now, if X_k and X_j are elevated by h_e, then we have $|HX_k| = h_i \frac{\sin\theta}{\sin(2\alpha + \theta)} + h_e$.

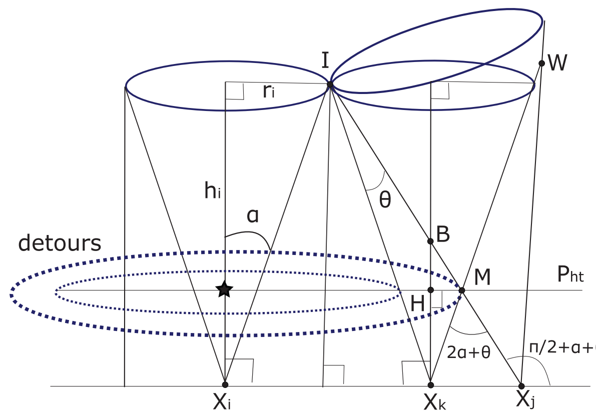

Fig. 12. If two cones C_j, C_k both intersect a third cone C_i, then they also intersect one another after a certain height h_t.

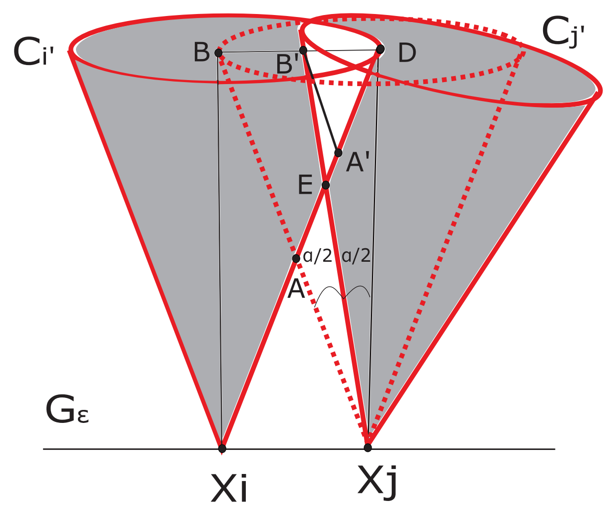

Fig. 13. Computing the volume of intersection between two cones $C_{i'}$, $C_{j'}$ with apex angle α, orientation angle difference $\theta = \epsilon_i \angle \epsilon_j \leqslant \frac{\alpha}{2}$, height h, and radius r. The volume of intersection between $C_{i'}$, $C_{j'}$ exists (gray area) and is always smaller than the volume from Lemma 4 by a constant factor.

Thus, if coverage height $h_t \geqslant hi \frac{\sin\theta}{\sin(2\alpha+\theta)} + h_e$, then these detours guarantee coverage of all cones intersecting C_i.

Lemma 10 implies that if two cones C_j, C_k with maximum orientation difference $\frac{\alpha}{2}$ both intersect a third cone C_i, then they also intersect one another after a certain height h_t (see Figure 12).

Lemma 11. *The additional detour from Algorithm 5 (Lemma 10) visits all cones not visited from Algorithm 3.*

Proof. In Figure 12, let cone c_k have the same height h_i and intersect cone $c_i \in MIS$ at its right-most cap point. The added detour from Lemma 10 visits c_k at its left-most and right-most cap points at coverage height h_t. Let cone c_j be tilted at an angle θ with respect to c_i and intersect c_i at its right-most cap point. Let W be the point of intersection between the two right-most rays of c_k and c_j. Triangle $\Delta X_k W X_j$ has $\hat{W} = \theta$ and $\hat{X}_k = \frac{\pi}{2} - \alpha$. From the law of sines we have $|X_j W| = |X_k X_j| \frac{\cos\alpha}{\sin\theta}$. From the proof of Lemma 10 we know $|X_k X_j| = \frac{r_i}{\sin\alpha} \frac{\sin\theta}{\cos(\alpha+\theta)}$. Combining the two we have $|X_j W| = \frac{h_i}{\cos(\alpha+\theta)}$. As $\frac{\pi}{2} > \theta > 0$, then $|X_j W| > \frac{h_i}{\cos(\alpha)}$ which is the length of the right-most ray of cone c_k. Also note that point $M \in c_i \in c_j$ and point $W \notin c_i \in c_j$ for $\theta > 0$. As points M, W belong to the same line $|MW| \in c_j$, it follows that the right-most ray of cone c_k above height $hi \frac{\sin\theta}{\sin(2\alpha+\theta)} + h_e$ is fully contained in cone c_j. Thus, the outermost detour always visits a tilted cone at distance greater than $4h_t \tan\alpha$ from the apex of c_i. We know from Lemma 8 that the innermost detours visit every other cone at distance $[0, 4h_t \tan\alpha]$, which includes any cone tilted at an angle θ closer to the apex of c_i than c_j.

In order to compute the performance of Algorithm 5 we need to revisit the lower bound from lemma 4. The following lemma shows how the different cone orientation angles affect the lower bound.

Lemma 12. *Let two cones C_i, C_j have the same height h_i, apex angle α, and orientation angle difference $\epsilon_i \angle \epsilon_j = \theta \leqslant \frac{\alpha}{2}$. A constant C_θ exists (similar to Lemma 4) such that the Minkowski sum sweep volume $f(T^*_G, \hat{h})$ is lower bounded by $f(g(T^*), h) \geqslant C_\theta \tan^2\alpha \sum_{h_i \in H} h_i^3$ where $C_\theta = C_\epsilon \frac{\tan^2\frac{\alpha}{2}}{(1-\tan\alpha\tan\theta)^5}$.*

Proof. In Figure 13, let line $B'A'$ originate from point B' and be parallel to BA. Let the area of $\Delta A'B'D$ be A_3, the area of ΔABD be A_1 and the area of $\Delta EB'D$ be A_2. The area of triangles ΔABD and $\Delta A'B'D$ are related by $\frac{A_3}{A_1} = \frac{|BD|}{|B'D|}^2 = \frac{\tan^2\frac{\alpha}{2}}{(1-\tan\alpha\tan\theta)^2}$. As $A_3 \leqslant A_2$ it follows $\frac{A_2}{A_1} \geqslant \frac{\tan^2\frac{\alpha}{2}}{(1-\tan\alpha\tan\theta)^2}$. Finally, note that the ratios of the volume between two cones C_1, C_2 and the areas of their maximum inscribed triangles T_1, T_2 is related by $\frac{vol(C_1)}{vol(C_2)} = \frac{1}{(1-\tan\alpha\tan\theta)^3} \frac{area(T_1)}{area(T_2)}$. Thus, the volume of intersection of cones $C_{i'}$, $C_{j'}$ compared with the volume of intersection of cones C_i, C_j from Lemma 4 is not smaller than $\frac{\tan^2\frac{\alpha}{2}}{(1-\tan\alpha\tan\theta)^5}$. □

Theorem 13. *Let h^* be the maximum height the optimal strategy achieves. For any set of input cones C with a given a coverage height $\hat{h}$ such that $\hat{h} \geqslant h_{max}$, $h^* \leqslant \hat{h} \leqslant 2h^*$ and orientation and apex angles ϵ and α such that $\epsilon \angle \epsilon_j = \theta \leqslant \frac{\alpha}{2}, \forall j$, $\theta + \alpha \in (0, \frac{\pi}{2})$, if the strategy Orientation-Visit solves Tilted Cone-TSPN using a $(1+\beta)$ approximation, then it has an approximation factor $(1+\beta)\left(\frac{18k}{\cos\epsilon}\left(1+\frac{5\pi\tan\alpha}{6}\right)\left\lfloor 1+\log_2\frac{\hat{h}}{h_{min}}\right\rfloor\right)$, where $k = \left\lceil \frac{2}{1-\tan\alpha\tan\epsilon} \right\rceil$.*

Proof. The proof is similar to Theorem 9. The added detours from Lemma 10 result in $k+1$ circumference paths each with length less than $8\pi h_t \tan\alpha$. These paths can be connected with a double line segment of size less than $4h_t \tan\alpha$. □

These bounds apply to cones having orientation angle difference $\theta \leqslant \frac{\alpha}{2}$. In case we have a wider range of cones

Algorithm 5. Orientation-Visit

Input: $C, x_0, H, \vec{A}, \epsilon, \alpha$
Output: Tilted Cone-TSPN tour T
for each orientation vector $\vec{a}_k \in \vec{A}$ **do**
  Create cone orientation set O_k with representative orientation $\vec{a}_k$
  Put all cones with orientation $\vec{a} \in \vec{A}$ such that $(\vec{a} \angle \vec{a}_k) \leqslant \frac{\alpha}{2}$ into set O_k
  Call Algorithm 3 with input O_k and perform $k+1$ detours
end for

with a larger orientation angle difference, we simply bin them into sets and perform the same strategy for each such set. Noting that $\hat{h} \leqslant max(H)$ and $\epsilon < \frac{\pi}{2} - \frac{\alpha}{2}$, we can simplify the approximation and obtain the result in Equation (1).

Lemma 14. *Given a set of orientation vectors $\vec{A}$, we need at most $\left\lceil 8\left(\frac{\pi}{\alpha}\right)^2 \right\rceil$ bins to separate them into sets of maximum angle difference of α.*

Proof. We will create a number of bins on the surface of a unit sphere such that for any pair of unit vectors $\vec{a}$, $\vec{b}$ on the same bin the condition $\vec{a} \angle \vec{b} \leqslant \alpha$ is satisfied. If $\vec{a}$ and $\vec{b}$ belong on a unit circle, they can be at most α away from each other and we need at least $\frac{2\pi}{\alpha}$ bins to cover all points of the circle. If $\vec{a}$ and $\vec{b}$ belong on a unit sphere, then we can divide it into a number of strips of width α, each centered around a circle that is equal to or smaller than the unit circle. As each strip is α away from each other, we can split the surface of the sphere into $\frac{2\pi}{\alpha}$ strips. We can cover each strip in its entirety with $2\frac{2\pi}{\alpha}$ bins and the entirety of the sphere with $8\frac{\pi^2}{\alpha^2}$ bins. □

8. Simulations

In Sections 4–7, we presented our strategy for the Tilted Cone-TSPN and its analysis for the worse-case performance. In this section, we present an implementation of our strategy and show through simulations that it can compute practical tours.

8.1. Implementation

We implemented our strategy along with heuristics that improve performance while keeping the theoretical guarantees. Instead of performing the entirety of all ellipsoidal detours on every cone in the *MIS*, we select a subset of them and only perform the part of the detour that visits another cone. In addition, we plan a tour with multiple orientation directions (azimuth angles). As performance depends on the relative cone arrangement, we consider multiple coverage heights and select the best. After identifying a set of points p_{tour} that cover all cones, we use the Concorde TSP solver (Applegate et al., 2006) in order to compute an optimized tour (see Figure 14 for an example tour).

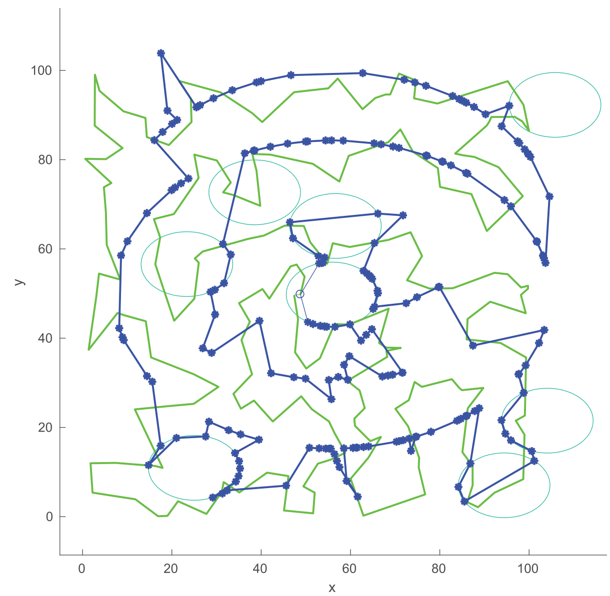

Fig. 14. An example of a computed tour, colored blue. For a set of randomly generated cones with apex angle $\pi/5$, tilt angle $\pi/4$ at coverage height 10 m. The ellipsoids represent the cones $\in MIS$. The computed tour (blue) is 26.94% better than the apex tour (green).

Our implementation is described in Algorithm 6. For each height guess h_t we define a horizontal plane P_{h_t} and compute the Tilted Cone-TSPN tour based on the ellipses resulting from the intersection of the cones with P_{h_t}. For each cone $c_i \in MIS$ we compute two ellipses $el1$ and $el2$ which correspond to the detours at radius $2h_t \tan\alpha$ and $2(k+1)(h_t - h_e)\tan\alpha$ on P_{h_t} (lines 4, 11). Then we identify the intersections between $el1, el2$ and the ellipses of cones $c_j \notin MIS$ (lines 7–14). We obtain at most two points for each cone c_j intersecting either $el1$ or $el2$ prioritizing the shorter detour $el1$. We keep the intersection point that has the closest neighbor among the currently selected set of points to be visited p_{tour} (lines 21–23). If a cone ellipse does not share any intersection points, then it is either on the MIS and does not intersect any cone or is not on the MIS. If it is not on the MIS, then it lies inside one of the ellipsoidal detours. In both cases, we choose to visit the cone at the point on its ellipse that is closest to another point $\in p_{tour}$ (lines 17–19). The optimal tour T_{h_t} visiting all points $\in p_{tour}$ is then computed using the Linkern module from the Concorde TSP solver (line 27). Finally, we select the best coverage height according to tour length (line 30).

8.2. Evaluation

We performed simulations for two representative applications (see also Section 2).

- The first application is coverage of reflective surfaces (see Figure 16). In this application, we select a number of view cones that cover a given square area and select a tilt angle ϵ that avoids direct sunlight (see also

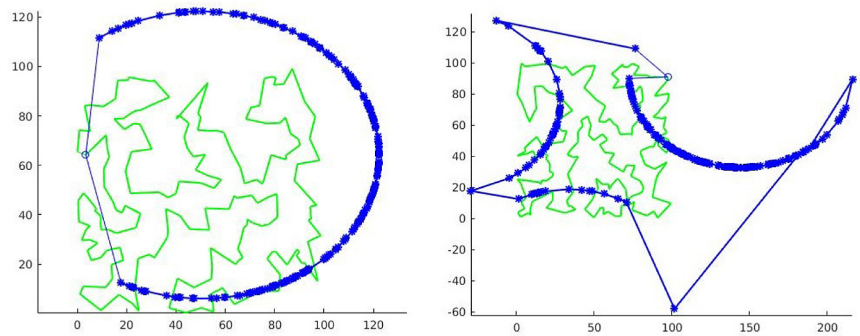

Fig. 15. The performance of Orientation-Visit-Practical depends on the cone arrangement. For 200 cones over a 100×100 m^2 area with an apex angle of $\frac{\pi}{5}$, cone tilting angle $\frac{\pi}{5}$, and azimuth angles $\{0, \pi/2, \pi, -\pi/2\}$, at a height of 80 m the computed tour on the left is 18.71% better than the apex tour (worse than any other coverage height). For a different cone arrangement the computed tour on the right is 63.05% better than the apex tour (best among all coverage heights).

Table 1. Average performance improvement of our strategy over the cone apex tour for sets of varying coverage heights $h_t = \{10, 20, 40, 80\}$ and cone tilting angles $\{\pi/4, \pi/5, \pi/10\}$. The second column of each tilt angle adds the cost of reaching coverage height h_t for the same tour as the first column. Cones have the same orientation vector direction (azimuth angle).

	Tilting angles & height cost					
	$\pi/4$ rad		$\pi/5$ rad		$\pi/10$ rad	
Coverage heights	$+0h_t$	$+2h_t$	$+0h_t$	$+2h_t$	$+0h_t$	$+2h_t$
10 m	27.60%	25.74%	25.99%	24.12%	24.70%	22.84%
20 m	42.01%	39.28%	43.58%	39.84%	41.53%	37.81%
40 m	61.80%	54.38%	61.72%	54.25%	59.60%	52.15%
80 m	63.05%	48.18%	65.90%	50.96%	68.90%	54.01%

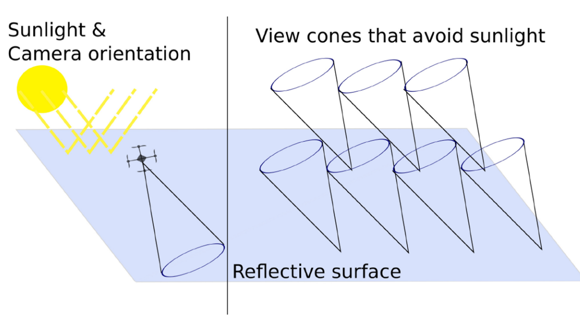

Fig. 16. Illustration of reflective surface coverage application. Given the sunlight direction we can calculate the camera orientation with the least amount of sunlight in the images. This results in view cones with uniform orientation.

Section 2.1). All cones have uniform tilt angle ϵ and orientation vector $\vec{a}$ direction (azimuth angle). The results can be seen in Table 1.

- The second application is visual inspection of a target area (see Figure 17). In this application, we select a number of side view cones that inspect a target area from all angles (see also Section 2.2). All cones share uniform tilt angle ϵ. However, the orientation vector $\vec{a}$ direction (azimuth angle) varies (non-uniform). The results can be seen in Table 2.

For both scenarios, we generated sets of 200 view cones with an apex angle of $\frac{\pi}{5}$. The apex points were positioned uniformly with $U(0, 100)$ over a 100×100 m^2 area. In total, 12 sets were generated with varying coverage heights $\in \{10, 20, 40, 80\}$ and cone tilt angles $\in \{\frac{\pi}{4}, \frac{\pi}{5}, \frac{\pi}{10}\}$. The azimuth angles for the first and second application were chosen $\in \{0\}, \{0, \frac{\pi}{2}, \frac{-\pi}{2}, \pi\}$, respectively. To evaluate the efficiency of our implementation we compare it with the TSP tour T_G that covers all the apex points on the ground. This tour, the cone apex tour, does not use any information about the cones and can be used as an upper bound on the length of any TSPN tour on our view cone problem.

The performance evaluation was based on the average tour length ratio between our implementation and the cone apex tour over 100 simulations. Tables 1 and 2 present how much shorter our tour is when compared with the cone apex tour for different coverage heights (columns) and cone tilt angles (rows). As the coverage height h_t increases the tour tends to get shorter owing to the increase in the ellipse area (first column for each tilt angle). However, the cost of reaching this height $+2h_t$ can make the total tour long and, thus,

Table 2. Average performance improvement of our strategy over the cone apex tour for sets of varying coverage heights $h_t = \{10, 20, 40, 80\}$ and cone tilting angles $\{\pi/4, \pi/5, \pi/10\}$. The second column of each tilt angle adds the cost of reaching coverage height h_t for the same tour as the first column. Cones have varying orientation vector directions (azimuth angles) $\in \{0, \pi/2, \pi, -\pi/2\}$.

	Tilting angles & height cost					
	$\pi/4$ rad		$\pi/5$ rad		$\pi/10$ rad	
Coverage heights	$+0h_t$	$+2h_t$	$+0h_t$	$+2h_t$	$+0h_t$	$+2h_t$
10 m	21.79%	19.93%	21.86%	19.99%	22.89%	21.03%
20 m	31.12%	27.38%	32.39%	28.66%	37.08%	33.36%
40 m	33.05%	25.59%	41.34%	33.84%	55.91%	48.47%
80 m	41.56%	26.62%	50.44%	35.51%	62.24%	47.35%

Algorithm 6. Orientation-Visit-Practical

Input: $C, x_0, H, \vec{A}, E, \alpha, X$
Output: Tilted Cone-TSPN tour T
Compute the TSP cone apex tour T_G visiting the apex points X
for each height guess h_t described in Algorithm 2 **do**
  Define horizontal plane P_{h_t} for height h_t as in Algorithm 1
  Intersect all cones $\in C$ with P_{h_t} and obtain the associated ellipse set ELC_{h_t}
  Sort ellipses into a set ELC_{sort} from shortest to tallest based on cone height
  $MIS_1 = MIS_2 = \emptyset$
  repeat
    Select the first ellipse $el1 \in ELC_{sort}$ (shortest cone in the set)
    $MIS_1 = MIS_1 \cup el1$
   Remove from ELC_{sort} ellipse $el1$ and all ellipses intersecting with it
   Create another ellipse $el2$ corresponding to the $k+1$ detour on P_{h_t}
   $MIS_2 = MIS_2 \cup el2$
   Remove from ELC_{sort} ellipse $el2$ and all ellipses intersecting with it
  until $ELC_{sort} = \emptyset$
  $p_{tour} = x_0$
  for each ellipse $el_i \in MIS_1 \cup MIS_2$ **do**
   if el_i does not intersect any other ellipse **then**
    Select the point $p_i \in el_i$ that is closest to another point $\in p_{tour}$
    Add p_i to p_{tour}
   else
    for each ellipse $el_j \in EL_{h_t}$ intersecting el_i **do**
     Select the intersection point p_{ij} that has the closest neighbor $\in p_{tour}$
     Add p_{ij} to p_{tour}
    end for
   end if
  end for
  Compute the TSP tour T_{h_t} visiting all points $\in p_{tour}$
  Add to T_{h_t} a vertical line segment of length $2h_t$ connecting P_{h_t} with G at x_0
end for
Select the best tour among all T_{h_t} and T_G

we need to consider multiple coverage heights and choose the best (second column for each tilt angle). The performance of each individual tour depends on the relative cone arrangement. The maximum performance difference was 44.34% (see Figure 15). These results show that our algorithm can be used in a practical way and provide shorter tours by exploiting the structure of the cones.

9. Conclusion and future work

In this work, we have studied the optimization problem Tilted Cone-TSPN, which is an extension of Cone-TSPN. We have demonstrated through field experiments that tilted view cones are necessary in real-world applications. Our main contribution is a polynomial-time approximation algorithm that solves Tilted Cone-TSPN and guarantees a solution that only depends on the cone apex angle α, tilt angle ϵ, and the ratio between the shortest and tallest cone. In addition to presenting the mathematical bounds, we implemented our strategy with heuristics that improve performance without sacrificing the theoretical guarantees. Simulations over large numbers of cones indicated that our strategy produced a tour that was shorter than the tour on the cone apex points that did not exploit the cone structure.

There are multiple venues for future work. One of the main assumptions of this work is that the conic regions are given ahead of time. What if the cone heights are known, but the apex positions can change (e.g., coverage of a moving target)? Furthermore, the current strategy does not consider prioritization of the conic regions. What if some regions are preferred over others? It may be advantageous to first perform a quick, high-altitude coverage and then a more detailed, lower-altitude coverage of points of interest. Similarly, if changing the tilt angle of the camera is slow, then different angles may be preferred over others. We would also like to explore the extreme case where the cones touch the ground. A different method and analysis might be required for this case, and a ground robot collaborating with a UAV might be a more appropriate option. In

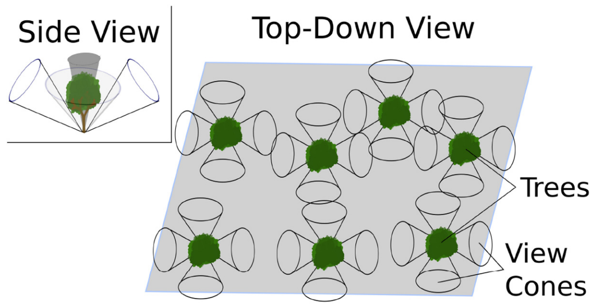

Fig. 17. Illustration of visual inspection of a target area application. In order to capture image footage of an occluded area (e.g., under trees), we require tilted view cones. This results in view cones that face in different directions (different azimuthal angles).

addition, multiple UAVs may be able to cover different height and orientation sets at the same time improving the coverage speed. Finally, we would like to determine whether a UAV can autonomously decide on the best coverage resolution for a given target and choose the best coverage height and orientation online.

Acknowledgements

A preliminary version of this article, was presented at the International Conference on Robotics and Automation (ICRA 2018). This submission extends the conference version by presenting the full details of the proofs, a revised implementation and field experiments. We thank Dr. Forest Isbell for allowing us to perform experiments at Cedar Creek Ecosystem Science Reserve.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported in part by the NSF (award numbers 1525045, 1617718 & 1849107) and a grant from Minnesota State LCCMR Program.

ORCID iDs

Nikolaos Stefas https://orcid.org/0000-0002-4940-1716
Patrick A Plonski https://orcid.org/0000-0002-7978-4586

References

Applegate D, Bixby R, Chvatal V and Cook W (2006) Concorde TSP solver. Available at: http://www.tsp.gatech.edu/concorde/.

Arora S (1998) Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems. *Journal of the ACM* 45(5): 753–782.

Bodlaender HL, Feremans C, Grigoriev A, Penninkx E, Sitters R and Wolle T (2009) On the minimum corridor connection problem and other generalized geometric problems. *Computational Geometry* 42(9): 939–951.

Chan THH and Elbassioni K (2011) A QPTAS for TSP with fat weakly disjoint neighborhoods in doubling metrics. *Discrete & Computational Geometry* 46(4): 704–723.

Cheng P, Keller J and Kumar V (2008) Time-optimal UAV trajectory planning for 3D urban structure coverage. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2008 (IROS 2008)*. IEEE, pp. 2750–2757.

Choset H (2001) Coverage for robotics—a survey of recent results. *Annals of Mathematics and Artificial Intelligence* 31(1): 113–126.

de Berg M, Gudmundsson J, Katz MJ, Levcopoulos C, Overmars MH and van der Stappen AF (2005) TSP with neighborhoods of varying size. *Journal of Algorithms* 57(1): 22–36.

Dumitrescu A and Mitchell JS (2001) Approximation algorithms for TSP with neighborhoods in the plane. In: *Proceedings of the 12th Annual ACM-SIAM Symposium on Discrete Algorithms*. Philadelphia, PA: Society for Industrial and Applied Mathematics, pp. 38–46.

Dumitrescu A and Tóth CD (2015) Constant-factor approximation for TSP with disks. *arXiv preprint arXiv:1506.07903*.

Dumitrescu A and Tóth CD (2016) The traveling salesman problem for lines, balls, and planes. *ACM Transactions on Algorithms* 12(3): 43.

Elbassioni K, Fishkin AV and Sitters R (2006) On approximating the TSP with intersecting neighborhoods. In: *International Symposium on Algorithms and Computation*. Berlin: Springer, pp. 213–222.

Englot B and Hover FS (2013) Three-dimensional coverage planning for an underwater inspection robot. *The International Journal of Robotics Research* 32(9–10): 1048–1073.

Galceran E and Carreras M (2013) A survey on coverage path planning for robotics. *Robotics and Autonomous Systems* 61(12): 1258–1276.

Low KH, Dolan JM and Khosla P (2008) Adaptive multi-robot wide-area exploration and mapping. In: *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems, Vol. 1*. International Foundation for Autonomous Agents and Multiagent Systems, pp. 23–30.

Mitchell JS (1999) Guillotine subdivisions approximate polygonal subdivisions: A simple polynomial-time approximation scheme for geometric TSP, k-MST, and related problems. *SIAM Journal on Computing* 28(4): 1298–1309.

Mitchell JS (2007) A PTAS for TSP with neighborhoods among fat regions in the plane. In: *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Philadelphia, PA: Society for Industrial and Applied Mathematics, pp. 11–18.

Plonski PA and Isler V (2019) Approximation algorithms for tours of height-varying view cones. *The International Journal of Robotics Research* 38(2–3): 224–235.

Reinelt G (1994) *The Traveling Salesman: Computational Solutions for TSP Applications*. Berlin: Springer-Verlag.

Sadat SA, Wawerla J and Vaughan RT (2014) Recursive non-uniform coverage of unknown terrains for UAVs. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2014 (IROS 2014)*. IEEE, pp. 1742–1747.

Safra S and Schwartz O (2006) On the complexity of approximating TSP with neighborhoods and related problems. *Computational Complexity* 14(4): 281–307.

Singh A, Krause A, Guestrin C and Kaiser WJ (2009) Efficient informative sensing using multiple robots. *Journal of Artificial Intelligence Research* 34: 707–755.

Stefas N, Bayram H and Isler V (2016) Vision-based UAV navigation in orchards. *IFAC-PapersOnLine* 49(16): 10–15.