2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

On Shortest Arc-To-Arc Dubins Path

Satyanarayana G. Manyam[1] and David W. Casbeer[2]

***Abstract*— For a given set of orbits, the Orbiting Dubins Traveling Salesman Problem (ODTSP) involves finding Dubins tour that is tangential to each orbit at some point. We consider a shortest Arc-to-Arc Dubins (ATAD) path problem that arrives in solving lower bound to the ODTSP. Given an initial and a final arc, the objective of ATAD is to find the shortest Dubins path such that the initial and final point lie on the given two arcs, and the path is tangential to the arcs. We analyze the six Dubins modes and the degenerate cases to find local minima. We present the optimal solution for the ATAD, along with an algorithm that uses this solution to compute tight lower bounds for the ODTSP. We test the lower bounding algorithm on several random instances and report the results. Using this algorithm, we show that the percent gap between upper and lower bounds is less than 10% for most instances.**

I. Introduction

The applications of fixed-wing aerial robots for surveillance and data gathering has been a topic of keen interest, and the routing problem for such applications are modeled as Dubins Traveling Salesman Problem (DTSP) [1]–[3]. DTSP involves finding the shortest tour that visits a given set of target locations, such that curvature constraints (equivalently minimum turn radius constraints) are satisfied at every point along the resulting path. A fundamental result needed to solve the DTSP is to find the shortest curvature constrained path between given initial and final positions and headings; this was originally solved by Dubins in 1957 [4], and an alternate proof using Pontryagin's minimum principle is presented in [5]. The Dubins paths are further analyzed in [6], [7]. The curvature constrained path planning problems with moderate obstacles were addressed in [8], [9].

Regarding the larger DTSP, there exists several classes of solutions. For example approximation algorithms are presented in [2] and [3], a gradient based approach is presented in [1], and several heuristic approaches are developed in [10]–[12]. For heuristic approaches, it is important to corroborate their performance, which can be accomplished by finding tight lower bounds with respect to the optimal solution. Towards this end, a partitioning approach developed in [13], [14] generates a tight lower bound by solving the Dubins Interval Problem (DIP). As inferred from its name, the DIP involves finding the shortest Dubins path between two locations for a given interval of headings (rather than a specific heading) at the start and final location.

[1]Satyanarayana G. Manyam is with Infoscitex Corporation, Dayton, OH, 45431, USA msngupta@gmail.com

[2]David Casbeer is with the Control Science Center, Air-Force Research Laboratories, WPAFB, OH, 45431, USA david.casbeer@us.af.mil

Distribution Statement A: Approved for Public Release; Distribution is Unlimited. PA AFRL-2022-4455

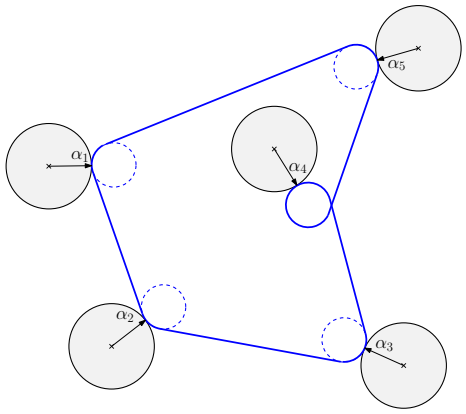


Fig. 1. A sample feasible solution of an ODTSP with 5 orbits. The tour (shown in blue) is tangential to the orbits at positions α_i's.

Alternatively, instead of modeling a location as a point, suppose the target location falls inside some prescribed area. Such a scenario is addressed by formulating the DTSP with Neighborhoods (DTSPN). This was addressed for general neighborhoods in [15], [16], and for the circular neighborhoods in [17]–[19]. Tight lower bounds for DTSPN are found using a generalization of Dubins Interval Problem, named Generalized Dubins Interval Problem (GDIP) [18].

Another key DTSP variant is the Orbiting Dubins Traveling Salesman Problem (ODTSP), where a set of orbits are prescribed, and the objective is to find the shortest tour that visits each orbit once. As shown in Fig. 1, the tour/path should be tangential to each orbit at some point, which differentiates it from DTSPN, where the tour has to visit at last one point in each neighborhood, and there are no constraints on the tangency. The ODTSP models robotic vehicles that are equipped with side facing sensor, which requires the vehicle to orbit around a target to sense it. More specifically for fixed-wing aerial robots with a gimballed sensor or underwater robotics, orbiting is also a desirable behavior as it maintains a fixed distance from the target. When the robots are equipped with sensor on only one lateral side, then orbiting direction (clockwise or counter-clockwise) has to be same at every target. For simplicity, we can assume the orbiting time required at every target is equal. Therefore, finding the shortest tour requires finding the sequence of orbits to be visited along with the entry/exit positions on each orbit. Variants of this problem were addressed without obstacles in [20]–[22], and with obstacles in [23]. As stated previously, a weakness in these prescribed heuristic algorithms is the lack of performance bounds. There are no algorithms that finds optimal solution

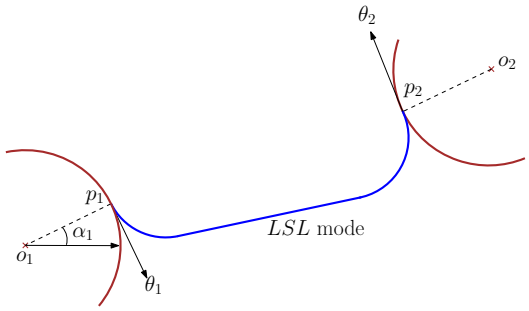


Fig. 2. An example path of ATAD, where positions p_1 and p_2 are chosen at the first and second arcs, and θ_1 and θ_2 are the corresponding headings.

to the ODTSP. Therefore, tight lower bounds would be useful for corroborating the performance of these approaches.

The partitioning approach in [13], [14], [24] produces tight lower bounds for DTSP and DTSPN and motivates a similar approach for the ODTSP. To this end, we introduce the Shortest Arc-to-Arc Dubins (ATAD) path problem. For a given initial and final arc, the objective of ATAD is to find the shortest Dubins path where the start and final positions are on the initial and final arc, and the headings at those positions are tangential to the corresponding arcs. A sample Dubins path following an LSL (Left turn - Straight line - Left turn) mode is shown in Fig. 2. A special case of ATAD, where the starting configuration is given, and the second arc is a circle was addressed in our earlier work [25]. Also, the distances between the starting configuration and the circle were assumed to be sufficiently large. ATAD generalizes [25], where both the initial and final configurations are constrained to lie on arcs, and no assumptions were made on the distance between the initial and final arc.

The contributions of this paper are the following: (i) We introduce and formulate the Arc-to-Arc Dubins (ATAD) path problem. (ii) We present the analysis of each Dubins mode and the optimal solution to the ATAD. (iii) We use the optimal solution of the ATAD to find tight lower bounds to the ODTSP, and test its performance on random instances with respect to the percent gap between upper and lower bounds. Due to space limitations, the analysis in this paper is restricted to clockwise orbits; however, the analysis for counter-clockwise orbits follows similarly.

The rest of the paper is organized as follows: The analysis of each of the Dubins modes is presented in Section II. This includes the analysis of the three segment modes, and the degenerate modes, where the length of one or more segments could be zero. Section III presents the computation of tight lower bound to ODTSP using the optimal solution of ATAD. Finally, conclusions are made in Section IV.

II. Analysis of Dubins Path

The configuration space of the robot is a subset of $\mathbb{R}^2 \times [0, 2\pi)$, and a configuration of a robot, denoted by q, defines its position and heading. Let ρ represent the minimum turn radius of the robot, or equivalently $\frac{1}{\rho}$ is the maximum curvature of the robot's path. The Dubins path is the shortest path between two configurations q_1 and q_2 that satisfies the curvature constraints, and its length is denoted as $(l_D(q_1, q_2))$. The Dubins path is the minimum of six modes in the set, $\mathcal{M} = \{LSL, LSR, RSR, RSL, LRL, RLR\}$, where L and R represent left and right turns of radius ρ, and S represents a straight line.

Let $\mathcal{A}_i$ represent an arc defined by the center o_i and a closed intervals of angular positions measured counter-clockwise $[\alpha_i^l, \alpha_i^u]$. The robot is restricted to visit an arc in clockwise direction, and therefore, an angular position $\alpha_j \in [\alpha_i^l, \alpha_i^u]$ uniquely defines a configuration $q_i(\alpha_j)$. Given two arcs $\mathcal{A}_1$ and $\mathcal{A}_2$, the Arc-to-Arc Dubins path aims to find the shortest Dubins path from a configuration $q_1(\alpha_1)$ to $q_2(\alpha_2)$, such that $\alpha_1 \in [\alpha_1^l, \alpha_1^u], \alpha_2 \in [\alpha_2^l, \alpha_2^u]$. The ATAD path could be stated as the following,

$$\min_{\alpha_1 \in [\alpha_1^l, \alpha_1^u], \alpha_2 \in [\alpha_2^l, \alpha_2^u]} l_D(q_1(\alpha_1), q_2(\alpha_2)).$$

For a given mode, $M \in \mathcal{M}$, $l_M(q_1, q_2)$ represents the length of the path of Dubins mode M from q_1 to q_2. To find the shortest Arc-to-Arc Dubins path the following logic is followed: for each Dubins mode $M \in \mathcal{M}$, determine the starting and ending positions yielding the shortest length path. The minimum length path out of each of these modes is then the shortest Arc-to-Arc Dubins path. Let $CSC_{\min}$ be the shortest CSC path from $\mathcal{A}_1$ to $\mathcal{A}_2$, and other modes are similarly defined.

Theorem 1. *The shortest Arc-to-Arc Dubins path from $\mathcal{A}_1$ to $\mathcal{A}_2$ is the path of minimum length in the set $\{CSC_{\min}, CCC_{\min}, CS_{\min}, SC_{\min}, CC_{\min}, C, S\}$.*

This minimum of each class of the paths are stated and proved in the rest of this Section. The analysis will start in Section II-A with turn-straight-turn, denoted as CSC, paths. This includes the LSL, LSR, RSR and RSL paths. Then, we proceed to turn-turn-turn, denoted CCC, paths (LRL and RLR) in Section II-B. The degenerate cases, which include C, S, CS, SC and CC paths, are discussed in Sections II-C through II-F.

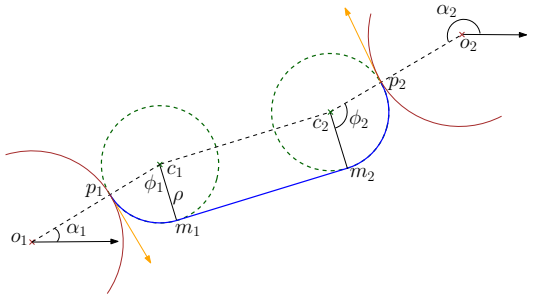
A. Dubins Mode: CSC

Notation: $\mu(A)$ is equal to 1, if A is true and is equal to -1, otherwise. δ_{ij} is equal to 1, if $i = j$, and is equal to 0, otherwise. For brevity, the partial derivative of a quantity f is written as $(f)_\alpha := \frac{\partial f}{\partial \alpha}$. Angle between two vectors, u and v, in counter clockwise direction from u to v is denoted as $\theta_{u,v}$.

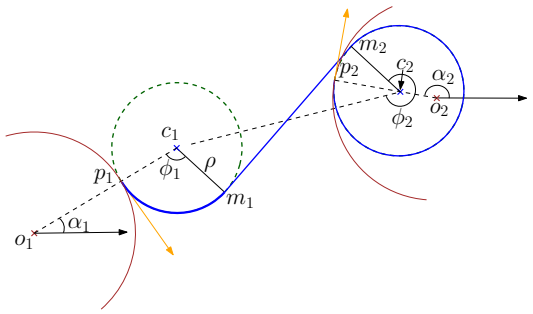
Property 1. *If u, v and w are three vectors such that either $u \cdot v = 0$ or $v \cdot w = 0$, then $u \cdot w = -\|u\| \|w\| \sin(\theta_{u,v}) \sin(\theta_{v,w})$.*

The above property is evident from the fact that $u \cdot w = \|u\| \|w\| \cos(\theta_{u,w})$, and we can write θ_{uw} as $\theta_{u,v} + \theta_{v,w}$. In the following analysis, we assume none of the segments of a CSC path vanishes, and analyze the degenerate two segment and one segment paths in later sections.

To find the minimum CSC path, we present the derivations and proofs for the LSL and LSR paths. For brevity and to avoid repetition, we omit the other two CSC modes, RSR and RSL, as those proofs follow a similar pattern.



(a) LSL path from $\mathcal{A}_1$ to $\mathcal{A}_2$



(b) LSR path from $\mathcal{A}_1$ to $\mathcal{A}_2$

Fig. 3. Dubins modes LSL and LSR from $p_1 \in \mathcal{A}_1$ to $p_2 \in \mathcal{A}_2$

Lemma 1. *The partial derivative of the length of a CSC path w.r.t. α_j, $\frac{\partial l_{CSC}}{\partial \alpha_j}$, is equal to*

$$-\mu(_{j=1})\mu(_{C_j=L})\rho + \mu(_{j=1})(r_j + \mu(_{C_j=L})\rho)\cos\phi_j,$$

and it vanishes when the straight line segments is collinear with o_j.

Proof. Let ϕ_1 and ϕ_2 represent the arc angles subtending first and second arcs of a CSC path, as shown in Figs. 3(a) and 3(b). For $i = 1, 2$, the arc angle ϕ_i and its partial derivative w.r.t. α_j are given as follows,

$$\phi_i(\alpha_1, \alpha_2) = \frac{1}{\rho^2}\arccos(\overline{c_i m_i}\cdot\overline{c_i p_i}) \mod 2\pi,$$
$$\frac{\partial \phi_i}{\partial \alpha_j} = -\frac{1}{\rho^2 \sin\phi_i}\left[\overline{c_i m_i}\cdot(\overline{c_i p_i})_{\alpha_j} + \overline{c_i p_i}\cdot(\overline{c_i m_i})_{\alpha_j}\right]. \quad (1)$$

Note that the vector $\overline{c_i p_i} = \rho[\cos\alpha_i, \sin\alpha_i]$, and therefore $\left\|(\overline{c_i p_i})_{\alpha_j}\right\| = \rho\delta_{ij}$. Using property 1, we have the following:

$$\overline{c_i m_i}\cdot(\overline{c_i p_i})_{\alpha_j} = -\rho^2\delta_{ij}\mu(_{i=1})\mu(_{C_i=R})\sin\phi_i. \quad (2)$$

Here, $\mu(_{C_i=R})$ is equal to 1 if the i^{th} arc is a right turn, and is -1 for left turn. Similarly, using the property 1, we get the following,

$$(\overline{c_i m_i})_{\alpha_j}\cdot\overline{c_i p_i}\cdot = -\rho\mu(_{i=1})\mu(_{C_i=R})\left\|(\overline{c_i m_i})_{\alpha_j}\right\|\sin\phi_i\sin\left(\theta_{(\overline{c_i m_i})_{\alpha_j},\overline{c_i m_i}}\right). \quad (3)$$

Substituting (2) and (3) in (1), we get

$$(\phi_i)_{\alpha_j} = \mu(_{i=1})\mu(_{C_i=R})\left[\delta_{ij} + \frac{\left\|(\overline{c_i m_i})_{\alpha_j}\right\|}{\rho}\sin\left(\theta_{(\overline{c_i m_i})_{\alpha_j},\overline{c_i m_i}}\right)\right]. \quad (4)$$

The straight line segment of the CSC path is $\overline{m_1 m_2} = \overline{o_2 c_2} + \overline{c_2 m_2} - \overline{o_1 c_1} - \overline{c_1 m_1} + \overline{o_1 o_2}$.

For an LSL path, $\overline{c_1 m_1} = \overline{c_2 m_2}$, and the partial derivative $(\overline{m_1 m_2})_{\alpha_j} = (\overline{o_2 c_2} - \overline{o_1 c_1})_{\alpha_j}$. The partial derivative, $(\overline{o_i c_i})_{\alpha_j}$ is equal to 0 if $i \neq j$, gives

$$(\overline{m_1 m_2})_{\alpha_j} = -\mu(_{j=1})(\overline{o_j c_j})_{\alpha_j}. \quad (5)$$

Let l_S be the length of the straight line segment, $l_S^2 = \overline{m_1 m_2}\cdot\overline{m_1 m_2}$. The partial derivative of l_S is $(l_S)_{\alpha_j} = u_{12}\cdot(\overline{m_1 m_2})_{\alpha_j} = -\mu(_{j=1})u_{12}\cdot(\overline{o_j c_j})_{\alpha_j}$, where u_{12} is the unit vector along $\overline{m_1 m_2}$. Using property 1,

$$(l_S)_{\alpha_j} = \mu(_{j=1})(r_j + \rho)\cos\phi_j. \quad (6)$$

Using (1), we get $\rho(\phi_1 + \phi_2)_{\alpha_j} = -\mu(_{j=1})\rho$. The length of the LSL path is $l_S + \rho(\phi_1 + \phi_2)$, and thus the partial derivative is obtained as following:

$$\frac{\partial l_{LSL}}{\partial \alpha_j} = \mu(_{j=1})\left[(r_j + \rho)\cos\phi_j - \rho\right]. \quad (7)$$

For an LSR path, $\overline{c_1 m_1} = -\overline{c_2 m_2}$, and the partial derivative $(\overline{m_1 m_2})_{\alpha_j} = (\overline{o_2 c_2} - \overline{o_1 c_1} - 2\overline{c_1 m_1})_{\alpha_j}$, and it is as shown below,

$$(\overline{m_1 m_2})_{\alpha_j} = -\mu(_{j=1})(\overline{o_j c_j})_{\alpha_j} - 2(\overline{c_1 m_1})_{\alpha_j}. \quad (8)$$

Similar to the LSL case, we can use Property 1 to derive

$$u_{12}\cdot(\overline{c_1 m_1})_{\alpha_j} = \left\|(\overline{c_1 m_1})_{\alpha_j}\right\|\sin(\theta_{\overline{c_1 m_1},(\overline{c_1 m_1})_{\alpha_j}}).$$

By substituting (8), we get the derivative of the length l_S as shown below,

$$(l_S)_{\alpha_j} = \mu(_{j=1})(r_j + \mu(_{C_j=L})\rho)\cos\phi_j - 2\left\|(\overline{c_1 m_1})_{\alpha_j}\right\|\sin(\theta_{\overline{c_1 m_1},(\overline{c_1 m_1})_{\alpha_j}}). \quad (9)$$

Using (1), we get

$$\rho(\phi_1 + \phi_2)_{\alpha_j} = -\rho + 2\left\|(\overline{c_1 m_1})_{\alpha_j}\right\|\sin(\theta_{\overline{c_1 m_1},(\overline{c_1 m_1})_{\alpha_j}}). \quad (10)$$

Adding (9) and (10), we get

$$\frac{\partial l_{LSR}}{\partial \alpha_j} = -\rho + \mu(_{j=1})\left((r_j + \mu(_{C_j=L})\rho)\cos\phi_j\right). \quad (11)$$

From equations (7) and (11), the partial derivatives vanish when $|\cos\phi_j| = \frac{\rho}{r_j + \mu(_{C_j=L})\rho}$, which corresponds to the instances where the straight line segment of the CSC path is collinear with o_j. □

Let α_i^c correspond to a position on arc $\mathcal{A}_i$ such that the straight line (middle) segment of the CSC path from $\mathcal{A}_1$ to $\mathcal{A}_2$ is collinear with the center o_i of arc $\mathcal{A}_i$. Let α_i^b denote any angular position that corresponds to the boundary of $\mathcal{A}_i$, *i.e.*, α_i^l or α_i^u.

Lemma 2.

$$\min_{\alpha_1\in[\alpha_1^l,\alpha_1^u],\alpha_2\in[\alpha_2^l,\alpha_2^u]} l_{CSC}(\alpha_1,\alpha_2) = \min\{l_{CSC}(\alpha_1^c,\alpha_2^c), l_{CSC}(\bar{\alpha}_1^c,\alpha_2^b), l_{CSC}(\alpha_1^b,\bar{\alpha}_2^c), l_{CSC}(\alpha_1^b,\alpha_2^b)\}.$$

Proof. From Lemma 1, the minimum of the length of a CSC path may occur at the critical point (α_1^c,α_2^c) or at the boundary conditions.

Now, consider the set of paths where one independent variable, say α_1 is equal to the boundary value α_1^b. The lengths of these CSC paths is a function of one variable; using Lemma 1 with α_1 as constant, the minimum of this occurs at $\bar{\alpha}_2^c$, which corresponds to a position such that straight line segment of the CSC path is collinear with o_2. The local minimum of these set of paths occur at $(\alpha_1^b,\bar{\alpha}_2^c)$ or $(\bar{\alpha}_1^c,\alpha_2^b)$. The final set of critical points are where both variables are at the boundary $(\alpha_1^b,\alpha_2^b), \alpha_i^b \in \{\alpha_i^l,\alpha_i^u\}$. □

B. Dubins Mode: CCC

Let m_1 and m_2 be the two inflection points on a CCC path from $\mathcal{A}_1$ to $\mathcal{A}_2$. Let α_i^c correspond to a position on arc $\mathcal{A}_i$ such that the three points m_1, m_2 and o_i are collinear. Let α_i^b corresponds to the angular positions of the bounds of an arc α_i^l or α_i^u.We prove the following lemma for only the RLR mode, and the proof for LRL follows similarly.

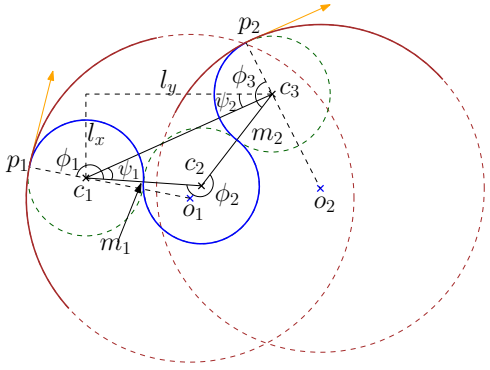


Fig. 4. Dubins mode RLR from $p_1 \in \mathcal{A}_1$ to $p_2 \in \mathcal{A}_2$

Lemma 3. *The partial derivatives of the length of a CCC path w.r.t α_i vanishes when the $\alpha_i = \alpha_i^c$.*

Proof. The difference in the angular position, α_1 and α_2, is a function of the arc angles ϕ_1, ϕ_2 and ϕ_3, $\alpha_1 - \alpha_2 = \phi_1 - \phi_2 + \phi_3$. The sum of the three arc angles is equal to $2\phi_2 + \alpha_1 - \alpha_2$. The partial derivative of the length of the RLR path is given as,

$$(l_{RLR})_{\alpha_j} = 2\rho(\phi_2)_{\alpha_j} + \mu_{(j=1)}\rho. \tag{12}$$

From Fig. 4, the arc angle, ϕ_2, is equal to $\pi + 2\psi_1$, where $\psi_1 = \arccos(\frac{l_{c_1c_3}}{4\rho})$. Here, $l_{c_1c_3} = \sqrt{l_x^2 + l_y^2}$, $[l_x, l_y]^T = o_1 + (r_1-\rho)[\cos\alpha_1 \sin\alpha_1]^T - o_2 - (r_2-\rho)[\cos\alpha_2 \sin\alpha_2]^T$. The partial derivative of ϕ_2 is computed as the following:

$$(\phi_2)_{\alpha_j} = -\mu_{(j=1)}\frac{1}{2\rho\sin\psi_1}(r_j-\rho)\sin(\psi_2-\alpha_j).$$

Substituting $(\phi_2)_{\alpha_j}$ in (12) gives the following:

$$(l_{RLR})_{\alpha_j} = \mu_{(j=1)}\left(-\frac{r_j-\rho}{\sin\psi_1}\sin(\psi_2-\alpha_j)+\rho\right). \tag{13}$$

For $j = 1, 2$, the partial derivatives, $(l_{RLR})_{\alpha_j}$, vanishes when $(r_j-\rho)\sin(\psi_2-\alpha_j) = \rho\sin\psi_1$. This implies that the inflexion points m_1 and m_2 are collinear with o_1 and o_2. □

We state the following lemma about the minimum CCC path, and the proof follows similar to the proof of Lemma 2.

Lemma 4.

$$\min_{\alpha_1\in[\alpha_1^l,\alpha_1^u],\alpha_2\in[\alpha_2^l,\alpha_2^u]} l_{ccc}(\alpha_1,\alpha_2) = \min\left\{l_{ccc}(\alpha_1^c,\alpha_2^c), l_{ccc}(\bar{\alpha}_1^c,\alpha_2^b), l_{ccc}(\alpha_1^b,\bar{\alpha}_2^c), l_{ccc}(\alpha_1^b,\alpha_2^b)\right\}.$$

C. Dubins Mode: S

This is a simple straight line path from $\mathcal{A}_1$ to $\mathcal{A}_2$, such that the straight line is clockwise tangent to $\mathcal{A}_1$ and $\mathcal{A}_2$. There always exists an outer tangent between two circles, and thus, an S path is feasible if the corresponding angular positions of tangency on the two circles lie in the given intervals $[\alpha_1^l,\alpha_1^u]$ and $[\alpha_2^l,\alpha_2^u]$, respectively. If no such path exists we set the length l_S to infinity.

D. Dubins Mode: C

This is a path from $\mathcal{A}_1$ to $\mathcal{A}_2$ that contains exactly one circular arc. Since we assume the orbiting direction is clockwise on $\mathcal{A}_1$ and $\mathcal{A}_2$, this path could be only a left turning arc. If no such path exists in the intervals $[\alpha_1^l,\alpha_1^u]$ and $[\alpha_2^l,\alpha_2^u]$, we set the corresponding length, l_C, to infinity.

E. Dubins Mode: CS or SC

The two segment paths from $\mathcal{A}_1$ to $\mathcal{A}_2$ have only one degree of freedom, *i.e.*, only one of the two angular positions, α_i's, can be an independent variable. The other is determined by the feasible two segment path. We analyse the characteristics to find the minimum of a CS path, and present the analysis of LS path. The analysis of the other three modes RS, SL and SR follows similarly. Let α_1^c and α_2^c corresponds to the positions on arcs $\mathcal{A}_1$ and $\mathcal{A}_2$, such that inflexion point of the LS path from $q_1(\alpha_1^c)$ to $q_2(\alpha_2^c)$ is collinear with centers of the arcs o_1 and o_2. Let α_1^b corresponds to the boundary positions on $\mathcal{A}_1$, and $\bar{\alpha}_1^b$ corresponds to the angular position on $\mathcal{A}_1$ such that $\alpha_2^b \in \{\alpha_2^l,\alpha_2^u\}$.

Lemma 5.

$$\min_{\alpha_1\in[\alpha_1^l,\alpha_1^u]} l_{CS}(\alpha_1) = \min\{l_{CS}(\alpha_1^c), l_{CS}(\alpha_1^b), l_{CS}(\bar{\alpha}_1^b)\}.$$

Proof. Without loss of generality we assume $o_1 = (0,0)$ and $o_2 = (d,0)$. Let l_S be the length of the straight line segment of the LS path, and let l_x and l_y be the x and y

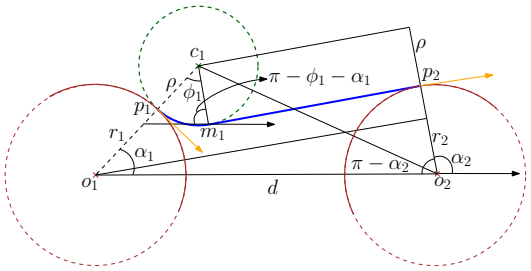


Fig. 5. Dubins mode LS from $\mathcal{A}_1$ to $\mathcal{A}_2$

coordinates of $\overline{c_1 o_2}$. Therefore, $l_S = \sqrt{l_x^2 + l_y^2 - (r_2+\rho)}$, where $l_x = d - (r_1+\rho)\cos\alpha_1$ and $l_y = -(r_1+\rho)\sin\alpha_1$.

$$(l_S)_{\alpha_1} = \frac{d(r_1+\rho)}{l_S}\sin\alpha_1. \tag{14}$$

We get the following two equations relating ϕ_1 and α_2,

$$(r_1+\rho)\sin\phi_1 + l_S = d\sin\alpha_2, \tag{15}$$
$$(r_1+\rho)\cos\phi_1 - d\cos\alpha_2 = (r_2+\rho). \tag{16}$$

We differentiate (15) and (16) w.r.t α_1 and solving for $(\phi_1)_{\alpha_1}$, we get $(\phi_1)_{\alpha_1} = -\frac{d}{l_S}\sin(\alpha_1+\phi_1)$. The partial derivative of the length of the LS path w.r.t. α_1 is equal to $(l_S)_{\alpha_1} + \rho(\phi_1)_{\alpha_1}$, simplified and shown below,

$$(l_{LS})_{\alpha_1} = \frac{d}{l_S}[(r_1+\rho)\sin\alpha_1 - \rho\sin(\alpha_1+\phi_1)]. \tag{17}$$

Clearly, from the Fig. 5, $(l_{LS})_{\alpha_1}$, vanishes when o_1, o_2 and m_1 are collinear. □

F. Dubins Mode: CC

Let α_1^c and α_2^c corresponds to the positions on arcs $\mathcal{A}_1$ and $\mathcal{A}_2$, such that inflexion point of the CC path from $q_1(\alpha_1^c)$ to $q_2(\alpha_2^c)$ is collinear with centers of the arcs o_1 and o_2. Let α_1^b and $\bar{\alpha}_1^b$ correspond to the boundary positions on $\mathcal{A}_1$ as defined in Section II-E.

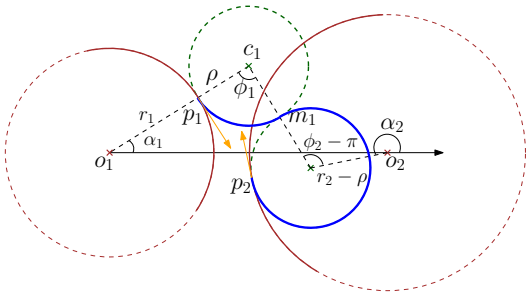


Fig. 6. Dubins mode LR from $\mathcal{A}_1$ to $\mathcal{A}_2$

Lemma 6.

$$\min_{\alpha_1 \in [\alpha_1^l, \alpha_1^u]} l_{CC}(\alpha_1) = \min\{l_{CC}(\alpha_1^c), l_{CC}(\alpha_1^b), l_{CC}(\bar{\alpha}_1^b)\}.$$

Proof. We prove this for the LR path and the proof for the RL follows similarly. We get the following equalities from Fig. 6:

$$(r_1+\rho)\cos\alpha_1 - 2\rho\cos(\alpha_1+\phi_1) - (r_2-\rho)\cos\alpha_2 = d,$$
$$(r_1+\rho)\sin\alpha_1 - 2\rho\sin(\alpha_1+\phi_1) - (r_2-\rho)\sin\alpha_2 = 0.$$

We derive $(\phi_1+\phi_2)_{\alpha_1}$ by differentiating the above two equations w.r.t. α_1,

$$\frac{\partial(\phi_1+\phi_2)}{\partial\alpha_1} = \left(1 + \frac{(r_1+\rho)\sin\alpha_1}{(r_2-\rho)\sin\alpha_2}\right)\left(\frac{(r_1+\rho)\sin(\alpha_2-\alpha_1)}{2\rho\sin\phi_2} - 1\right).$$

The derivative, $(\phi_1+\phi_2)_{\alpha_1}$, is equal to zero when $(r_1+\rho)\sin\alpha_1 = (r_2-\rho)\sin(\alpha_2-\pi)$, which corresponds to the instance where the inflexion point, m_1 is collinear with o_1 and o_2. □

III. Lower Bound to ODTSP

Given a set of orbits, the ODTSP aims to find the shortest Dubins tour that visits every orbit once. We assume the robotic vehicle needs to go one full orbit around each target. We assume the orbit radius around each target is the same, as it depends on the range sensor equipped on the robot. Since, the sum of the length of the orbits around the targets is the same for every sequence of visits, we do not include the orbiting length in the tour length. Therefore, an orbit is considered visited by a tour if it is tangential to a segment of the tour at some point. We refer to this point of tangency as the point of visit, and at each target i, it is uniquely defined by an angular position $\alpha_i \in [0, 2\pi)$. To find the shortest Dubins tour, the optimization problem needs to find the sequence of orbits to visit and the points of visit at each orbit. A feasible solution to this problem can be found by sampling methods as done in [26]. We uniformly sample the positions on each orbit for entry/departure, and pose a *Generalized TSP* (GTSP)[1], where each set contains the sampled positions of an orbit.

A lower bound to this problem could be computed using the partitioning approach as described below. In a feasible tour, the position at which a robot enters an orbit and departs the orbit are the same. We partition the orbit into set of arcs, and relax the continuity constraint, *i.e.*, the position of entry and departure could be different, however, they are restricted to belong to the same arc partition. This problem is posed as a GTSP where each set consists a set of arcs that belong to an orbit. This requires to find the shortest Dubins path from an arc in a set to an arc in another set. We compute this for every pair of arcs using the solution provided in Section II. The resulting GTSP is transformed to an asymmetric TSP and solved to optimality.[2]

One can tighten the bounds by using a more finely grained partition of the arcs. Doing so would require higher

[1]For a given sets of targets, GTSP aims to find the shortest tour such that a target is visited in each set.

[2]The resulting GTSP is transformed into a regular asymmetric TSP using the Noon and Bean transformation [27]. The asymmetric TSP is transformed into symmetric TSP [28] and is solved to optimality using the Concorde TSP solver [29].

computation time. The following iterative algorithm can be used to alleviate this increased computation time. In each iteration, the most promising arcs are partitioned; this is similar to the lower bounding of Dubins Touring Problem in [30]. Initially, the algorithm partitions each orbit into n_d arcs, and solves the GTSP to find a lower bound. In the iterative algorithm, the arc from each orbit that was selected by the GTSP solution is further partitioned and solves a new GTSP. This process is done iteratively until a stop criteria is met, which in this work is defined as a maximum number of iterations. The pseudocode is presented in Algorithm 1. As an input, the algorithm takes the set of orbits, $\mathcal{O}$, and discretization parameter, n_d, as the inputs. In steps 3-4, each orbit is partitioned into arcs. A GTSP is posed using this set of arcs and solved iteratively, shown in steps 6-7. The set of arcs chosen by the GTSP are denoted as T_A, and they are further partitioned in the step 9. Finally, when the maximum iterations are reached, the algorithm outputs the cost of the GTSP, which is a lower bound to the ODTSP.

Algorithm 1 Iterative Lower Bounding of ODTSP

1: **function** LOWERBOUNDODTSP($\mathcal{O}, n_d$)
2: $S_A \leftarrow$ INITIALIZEARCS
3: **for** $O_i \in \mathcal{O}$ **do**
4: $S_A \leftarrow S_A \cup$ PARTITION(O_i, n_d)
5: **while** $i < maxIters$ **do**
6: $S_O \leftarrow$ CONSTRUCTSETS(S_A)
7: $T_A \leftarrow$ GTSP(S_O)
8: **for** $A_k \in T_A$ **do**
9: $A_k \leftarrow$ PARTITION(A_k, n_d)
10: $S_A \leftarrow$ UPDATE(S_A, A_k)
11: $i \leftarrow i + 1$
12: **return** COST(T_A)

A feasible solution and solution generated by the lower bounding algorithm are shown in Fig. 7. Due to the relaxation of the continuity constraint, we can observe that the lower bounding path, shown in green, is discontinuous at the orbits. Also, the figure shows the arcs generated by the iterative sampling algorithm, and it is evident that algorithm generates fewer arcs in the less promising regions. The algorithm was tested on fifty random instances with 10 targets. The improvement of the gap over different iterations are shown as box and whisker plot in Fig. 8. Also shown is the computation time over the iterations. Clearly, the gaps reduce over the iterations, and the final gaps were less then 10% for most instances. The solution of DTSPN from [30] also gives a lower bound to the ODTSP. We use the sequence of orbits obtained from the proposed algorithm to find lower bound to the DTSPN, which in turn is a lower bound to the ODTSP. Fig. 9 compares percentage gap between upper and lower bounds using the proposed method and the lower bound of the DTSPN.

IV. CONCLUSION

We formulated the shortest Arc-to-Arc Dubins path problem, that is a key problem to be solved for computing tight

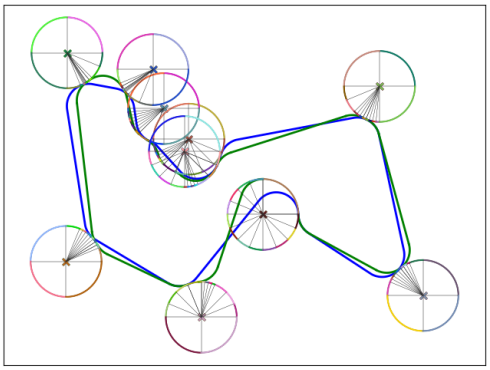


Fig. 7. A feasible solution (blue) and the lower bound solution (green) for an instance with 10 orbits. The arcs generated at each orbit are shown in different colors.

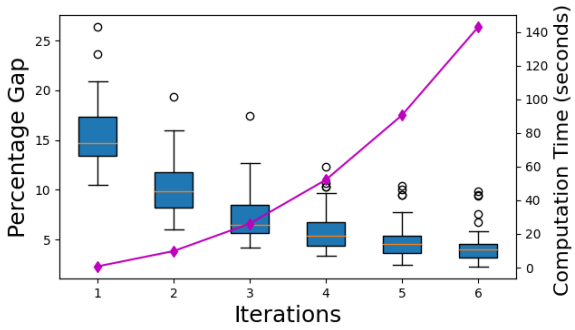


Fig. 8. Box and whisker plot of the percent gap between upper and lower bounds run for 50 randomly generated instances. Also shown is the plot of the mean computation time (on a different y-axis).

lower bounds to the Orbiting Dubins Traveling Salesman Problem. The six Dubins modes and their degenerate cases were analyzed to find the shortest Dubins path. We present the optimal solution for this problem, and used it to compute the lower bounds to the ODTSP, where the orbiting is restricted to clockwise direction. The lower bounding algorithm was run on several random instances, and were able to bring the gap between upper and lower bounds to less than 10% for most of the instances. As a future direction, we would generalize the ATAD and lower bounding algorithm to the general case where there are no restrictions on the orbiting direction. Another future direction is to construct feasible solutions using the lower bound solutions.

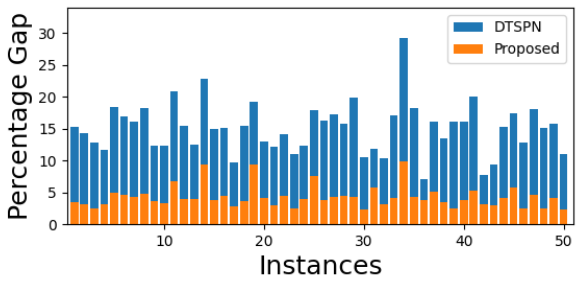


Fig. 9. Gap comparison between the proposed algorithm and the lower bound from solving the DTSPN

REFERENCES

[1] Z. Tang and U. Ozguner, "Motion planning for multitarget surveillance with mobile sensor agents," *IEEE Transactions on Robotics*, vol. 21, no. 5, pp. 898–908, 2005.

[2] S. Rathinam, R. Sengupta, and S. Darbha, "A resource allocation algorithm for multivehicle systems with nonholonomic constraints," *IEEE Transactions on Automation Science and Engineering*, vol. 4, no. 1, pp. 98–104, 2007.

[3] J. Ny, E. Feron, and E. Frazzoli, "On the Dubins traveling salesman problem," *IEEE Transactions on Automatic Control*, vol. 57, no. 1, pp. 265–270, 2012.

[4] L.E.Dubins, "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents," *American Journal of Mathematics*, vol. 79, no. 3, pp. 487–516, 1957.

[5] J.-D. Boissonnat, A. Cérézo, and J. Leblond, "Shortest paths of bounded curvature in the plane," *Journal of Intelligent and Robotic Systems*, vol. 11, no. 1, pp. 5–20, 1994.

[6] A. M. Shkel and V. Lumelsky, "Classification of the Dubins set," *Robotics and Autonomous Systems*, vol. 34, no. 4, pp. 179–202, 2001.

[7] X.-N. Bui, J.-D. Boissonnat, P. Soueres, and J.-P. Laumond, "Shortest path synthesis for Dubins non-holonomic robot," in *International Conference on Robotics and Automation*. IEEE, 1994, pp. 2–7.

[8] P. K. Agarwal, P. Raghavan, and H. Tamaki, "Motion planning for a steering-constrained robot through moderate obstacles," in *Proceedings of the twenty-seventh annual ACM symposium on Theory of Computing*, 1995, pp. 343–352.

[9] J.-D. Boissonnat and S. Lazard, "A polynomial-time algorithm for computing a shortest path of bounded curvature amidst moderate obstacles," in *Proceedings of the twelfth annual symposium on Computational geometry*. ACM, 1996, pp. 242–251.

[10] X. Ma and D. A. Castanon, "Receding horizon planning for Dubins traveling salesman problems," in *Proceedings of the 45th IEEE Conference on Decision and Control*, 2006, pp. 5453–5458.

[11] L. Babel, "New heuristic algorithms for the Dubins traveling salesman problem," *Journal of Heuristics*, vol. 26, no. 4, pp. 503–530, 2020.

[12] J. Faigl, P. Vana, and J. Drchal, "Fast sequence rejection for multi-goal planning with dubins vehicle," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 6773–6780.

[13] S. G. Manyam, S. Rathinam, D. Casbeer, and E. Garcia, "Tightly bounding the shortest Dubins paths through a sequence of points," *Journal of Intelligent & Robotic Systems*, vol. 88, no. 2, pp. 495–511, 2017.

[14] S. G. Manyam and S. Rathinam, "On tightly bounding the Dubins traveling salesman's optimum," *Journal of Dynamic Systems, Measurement and Control*, vol. 140, no. 7, p. 071013, 2018.

[15] K. J. Obermeyer, P. Oberlin, and S. Darbha, "Sampling-based path planning for a visual reconnaissance unmanned air vehicle," *Journal of Guidance, Control, and Dynamics*, vol. 35, no. 2, pp. 619–631, 2012.

[16] J. T. Isaacs and J. P. Hespanha, "Dubins traveling salesman problem with neighborhoods: A graph-based approach," *Algorithms*, vol. 6, no. 1, pp. 84–99, 2013.

[17] D. G. Macharet, A. A. Neto, V. F. da Camara Neto, and M. F. Campos, "Efficient target visiting path planning for multiple vehicles with bounded curvature," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2013, pp. 3830–3836.

[18] P. Vána and J. Faigl, "Optimal solution of the generalized Dubins interval problem." in *Robotics: Science and Systems*, 2018.

[19] R. Pěnička, J. Faigl, M. Saska, and P. Váňa, "Data collection planning with non-zero sensing distance for a budget and curvature constrained unmanned aerial vehicle," *Autonomous Robots*, vol. 43, no. 8, pp. 1937–1956, 2019.

[20] A. Wolek, J. McMahon, B. R. Dzikowicz, and B. H. Houston, "The orbiting dubins traveling salesman problem: planning inspection tours for a minehunting AUV," *Autonomous Robots*, vol. 45, no. 1, pp. 31–49, 2021.

[21] M. Couillard, J. Fawcett, and M. Davison, "Optimizing constrained search patterns for remote mine-hunting vehicles," *IEEE Journal of Oceanic Engineering*, vol. 37, no. 1, pp. 75–84, 2011.

[22] C. Hanson, J. Richardson, and A. Girard, "Path planning of a dubins vehicle for sequential target observation with ranged sensors," in *Proceedings of the 2011 American Control Conference*, 2011, pp. 1698–1703.

[23] H. Huang and A. V. Savkin, "Viable path planning for data collection robots in a sensing field with obstacles," *Computer Communications*, vol. 111, pp. 84–96, 2017.

[24] P. Váňa and J. Faigl, "On the Dubins traveling salesman problem with neighborhoods," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 4029–4034.

[25] S. G. Manyam, D. Casbeer, A. L. V. Moll, and Z. Fuchs, *Shortest Dubins path to a circle*.

[26] P. Oberlin, S. Rathinam, and S. Darbha, "Today's traveling salesman problem," *Robotics & Automation Magazine*, vol. 17, no. 4, pp. 70–77, 2010.

[27] C. E. Noon and J. C. Bean, "An efficient transformation of the generalized traveling salesman problem," *INFOR: Information Systems and Operational Research*, vol. 31, no. 1, pp. 39–44, 1993.

[28] R. Roberti and P. Toth, "Models and algorithms for the asymmetric traveling salesman problem: an experimental comparison," *EURO Journal on Transportation and Logistics*, vol. 1, no. 1-2, pp. 113–133, 2012.

[29] D. L. Applegate, R. E. Bixby, V. Chvátal, and W. J. Cook, *The Traveling Salesman Problem: A Computational Study*. Princeton: Princeton University Press, 2011. [Online]. Available: https://doi.org/10.1515/9781400841103

[30] J. Faigl, P. Váňa, M. Saska, T. Báča, and V. Spurný, "On solution of the Dubins touring problem," in *European Conference on Mobile Robots (ECMR)*, 2017, pp. 1–6.