AN APPROXIMATION ALGORITHM FOR A SHORTEST DUBINS PATH PROBLEM

Sivakumar Rathinam
Associate Professor
Department of Mechanical Engineering
Texas A & M University
College station, Texas, U.S.A, 77843

Pramod Khargonekar
Professor
Department of Electrical and Computer Engineering
University of Florida
Gainesville, Florida, U.S.A., 32611

ABSTRACT

The problem of finding the shortest path for a vehicle visiting a given sequence of target points subject to the motion constraints of the vehicle is an important problem that arises in several monitoring and surveillance applications involving unmanned aerial vehicles. There is currently no algorithm that can find an optimal solution to this problem. Therefore, heuristics that can find approximate solutions with guarantees on the quality of the solutions are useful. The best approximation algorithm currently available for the case when the distance between any two adjacent target points in the sequence is at least equal to twice the minimum radius of the vehicle has a guarantee of 3.04. This article provides a new approximation algorithm which improves this guarantee to $1 + \frac{\pi}{3} \approx 2.04$. The developed algorithm is also implemented for hundreds of typical instances involving at most 30 points to corroborate the performance of the proposed approach.

1 Introduction

Advances in sensing, robotics and wireless networks have enabled the use of teams of small Unmanned Vehicles (UVs) for environmental sensing applications including crop monitoring [1, 2], ocean bathymetry [3], forest fire monitoring [4], ecosystem management [5, 6], and civil security applications such as border surveillance [7, 8] and disaster management [9]. These applications frequently require vehicles to collect data such as visible/infra-red/thermal images, videos of specified target sites, and environmental data such as temperature, moisture, humidity using onboard sensors, and deliver them to a base station. To accomplish this requirement, small unmanned vehicles with motion and fuel constraints commonly visit a set of target points.

There are many fundamental problems that arise here which relate to path planning, control and navigation. This article addresses an important path planning problem that arises in a typical surveillance mission involving a fixed wing Unmanned Aerial Vehicle (UAV) with minimum turning radius constraints.

Given a sequence of target points to visit on a ground plane, this article considers the problem of finding a shortest path passing through the points in the given sequence such that the radius of curvature of any point on the path is at least equal to a positive constant. The curvature constraint imposed on the path models the minimum turning radius of a fixed wing UAV. If the vehicle is traveling at constant speed, this constraint also models the bound on the maximum yaw rate of a fixed wing UAV. Onboard resources such as fuel are limited for small vehicles and therefore, minimizing the length of a vehicle's path can lead to using the resources as efficiently as possible. This problem is referred to as the Curvature constrained Shortest path Problem (CSP) in this article (refer to figure 1). The CSP is also one of the two important subproblems of the well known Dubins Traveling Salesman Problem which has received significant attention in the literature [10–19].

The CSP is a generalization of the two point shortest path problem solved by L.E. Dubins [20] in 1957. Dubins [20] addressed the problem of finding a shortest path for a vehicle to travel from a point at (x_1, y_1) with heading angle θ_1 to a point at (x_2, y_2) with heading angle θ_2 such that the radius of curvature at any point along the path is at least equal to $\rho \geq 0$. Dubins [20] showed that any shortest path to this two point problem must consist of at most three segments where each of the segments must be an arc of radius ρ (denoted by C) or a straight line segment (denoted by S). Specifically, any shortest path is of type *CSC* or *CCC*, or a degenerate form of these paths. These paths
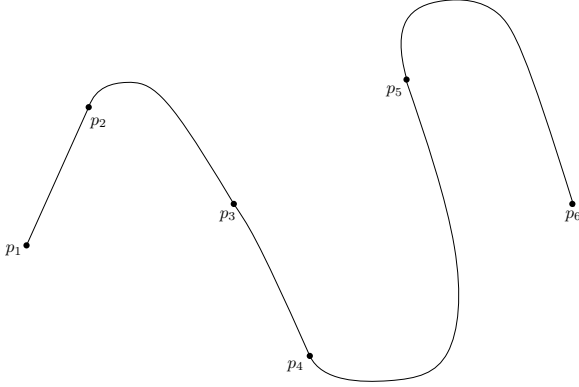
Figure 1. A possible feasible path for the CSP visiting a sequence of points denoted by $(p_1, \cdots, p_6)$.

are generally referred to as the Dubins paths in the literature. If the heading angle at each point is known, then the CSP simply reduces to finding an optimal Dubins path between any two adjacent target points in the given sequence. Therefore, solving the CSP requires us to find an optimal heading angle at each of the target points. This is a non-trival problem because the length of the CSP is a non-linear function of the heading angles at the target points.

There is currently no algorithm that can find an optimal solution to the CSP. Therefore, algorithms that can find feasible solutions with theoretical guarantees on the deviation of the solutions from the optimum are useful. The CSP was first considered by Lee et al. in [21] where they provide a 5.03-approximation algorithm. The approximation factor of this algorithm has been recently improved to $(2 + \frac{2}{\pi} + \frac{\pi}{2} \approx 4.21)$ in [22]. An α-approximation algorithm for a minimization problem is an algorithm that finds a feasible solution whose cost is at most α times the optimal cost for any instance of the problem. The factor α is also referred to as the approximation factor of the algorithm.

This article considers an important case of this problem where the distance between any two adjacent points is at least equal to 2ρ. In many practical applications, the sensors onboard a vehicle covers a wide swath of the monitoring area, and as a result, the vehicle may not be required to visit points that are close [19]. For this case, a 3.04-approximation algorithm was provided by Rathinam et al. in [19]. In this article, we improve on this bound and develop a new approximation algorithm with a factor of $1 + \frac{\pi}{3} + \varepsilon \approx 2.04 + \varepsilon$ for any small constant $\varepsilon > 0$.

2   Problem Statement

Consider a sequence of n points denoted by $(p_1, p_2, \cdots, p_n)$ on a plane. Let the position coordinates of the point p_i be denoted as (x_i, y_i). Without loss of generality, assume $n = 3k$ where k is any positive integer. The vehicle is first required to visit p_1, then p_2, and so on. The minimum turning radius of the vehicle is denoted as ρ. Let the Euclidean distance between any two adjacent points in the path visited by the vehicle be at least equal

to 2ρ. The objective of the CSP is to find a path such that the path visits each of the points in the order given by $(p_1, p_2, \cdots, p_n)$, the radius of curvature at any point along the path is at least equal to ρ and the length of the path is a minimum.

In the next section, we present a $(1 + \varepsilon)$-approximation algorithm for any given $\varepsilon > 0$ when $n = 3$. We then use this three point algorithm to develop an approximation algorithm for the general case in the subsequent section.

3 $(1 + \varepsilon)$-**Approximation algorithm for 3 points**

Let the Euclidean distance between points p_1 and p_2 be denoted as d_{12} and the Euclidean distance between points p_2 and p_3 be denoted as d_{23}. Without loss of generality, assume p_1 is at (x_1, y_1), p_2 is at the origin and p_3 is at $(d_{23}, 0)$. Also, the case when $x_1 \leq 0, y_1 = 0$ is not considered since the optimal solution in this case is just a straight line from p_1 to p_3. A curved segment C may either require the vehicle to turn in the clockwise direction (this curved segment with a right turn is denoted as R) or turn in the counter clockwise direction (this curved segment with a left turn is denoted as L).
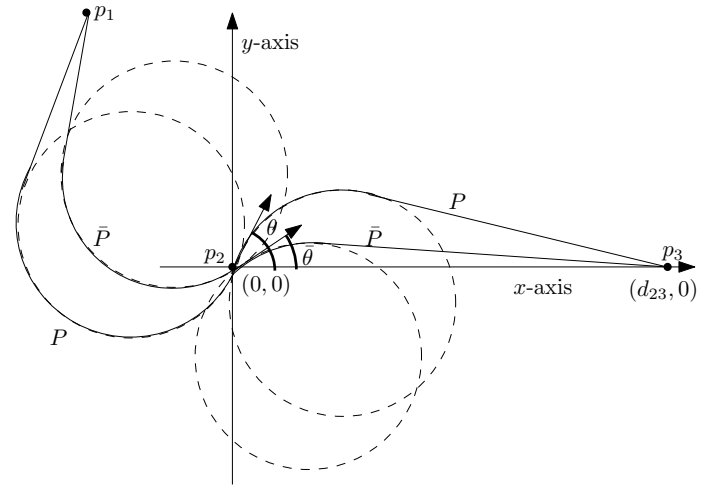

Figure 2. Let P denote a SL path from p_1 to p_2 and a RS path from p_2 to p_3. The heading angle θ corresponding to path P at p_2 can be reduced by a small amount to say $\bar{\theta}$ to obtain a new path $\bar{P}$ with a shorter length. The difference between the angles θ and $\bar{\theta}$ is magnified just for illustration.

Lemma 1. *The shortest path of bounded curvature through points p_1, p_2 and p_3 must be of type SRS if $y_1 < 0$, SLS if $y_1 > 0$ and either SRS or SLS if $y_1 = 0$. The point p_2 lies on the curved segment of the path.*
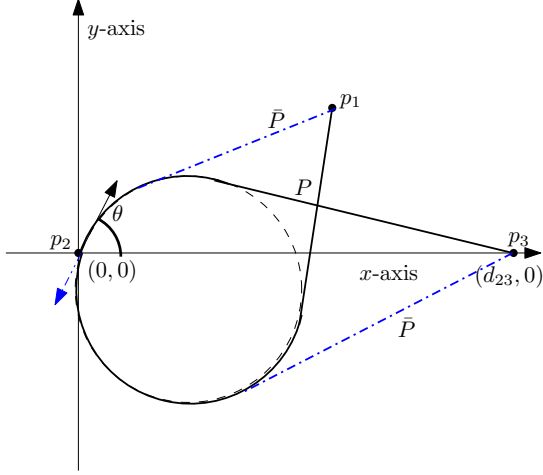
Figure 3. In this example $y_1 > 0$. P here denotes a SRS path from p_1 to p_3. The straight line segments of P cross each other when $y_1 > 0$. Therefore, *SRS* cannot be optimum because one can find a shorter path of type *SLS* where the straight line segments (dotted blue lines) do not cross each other. It is easy to verify that the length of the *SLS* path will be less than the length of the *SRS* path when $y_1 > 0$.

Proof. Given the heading angle θ at p_2, as $d_{12} \geq 2\rho$ and $d_{23} \geq 2\rho$, it is known [23] that the shortest path from p_1 to p_2 must be of type *SC* while the shortest path from p_2 to p_3 must be of type *CS*. For example, the path from p_1 to p_2 can be of type *SL* or *SR*. However, it is not optimal to arrive at p_2 in the counter clockwise direction and leave p_2 in the clockwise direction as shown in figure 2. Similarly, it is also not optimal to arrive at p_2 in the clockwise direction and leave p_2 in the counter clockwise direction. Therefore, *SCS* is the only possibility where the vehicle is turning either left or right in the curved segment while visiting p_2.

In addition, *SRS* cannot be optimal when the y coordinate of point p_1 is greater than zero (refer to figure 3). Similarly, *SLS* cannot be optimal when the y coordinate of point p_1 is less than zero. Hence proved. ∎

For the three point problem, if the heading angle at the first point is given, authors in [18] prove a similar result by formulating the shortest path problem as an optimal control problem and using Pontryagin's minimum principle. Apart from the difference in our approaches and the fact that we do not have any heading angle constraint at the first point, our work differs from the need to also understand the computational complexity of finding an optimal path. The next theorem shows that given any small $\varepsilon > 0$, the number of steps required to find an optimum is in the order of $\log_2 \frac{1}{\varepsilon}$ (which is a constant).

Theorem 1. *Given any $\varepsilon > 0$, a path within $(1 + \varepsilon)$ times the length of the shortest path can be found in polynomial time for three points.*

Proof. From lemma 1, the shortest path is of type *SLS* or *SRS*. Here, we show that a path of type *SRS* with length $(1 + \varepsilon)$ times the optimal length can be found in polynomial time when $y_1 < 0$. A similar proof can also be provided for *SLS*. Refer to figure 4 for an illustration of the *SRS* path when $y_1 < 0$ and all the notations for the involved angles. Note that the angles ϕ and ψ shown in figure 4 are functions of θ. Let $\mathfrak{D}_1$ be the length of the *SR* path from p_1 to p_2 as a function of the heading angle θ at p_2. Let $\mathfrak{D}_2$ be the length of the *RS* from p_2 to p_3 as a function of the heading angle θ at p_2. The function $\mathfrak{D}_1(\theta) + \mathfrak{D}_2(\theta)$ is discontinuous at $\theta = 0$ and $\theta = \beta$. At all other values of θ, both the length functions $\mathfrak{D}_1(\theta)$ and $\mathfrak{D}_2(\theta)$ are continuously differentiable [24]. Using the results for the derivatives of the length function of an *RS* path in [24], wherever the derivatives exist, we have,

$$\frac{d\mathfrak{D}_1}{d\theta} + \frac{d\mathfrak{D}_2}{d\theta} = -d_{12} \sin\psi + d_{23} \sin\phi.$$

$\frac{d\mathfrak{D}_1}{d\theta} + \frac{d\mathfrak{D}_2}{d\theta} = 0$ implies that the sum of the length functions can reach a minimum or a maximum when $d_{12} \sin\psi = d_{23} \sin\phi$. Now, $d_{12} \sin\psi$ reaches a maximum value of 2ρ when $\theta = \theta_{12}^*$ (refer to figure 5) and reduces to zero as θ is further increased to β. Without loss of generality, assume that $-\pi \leq \theta_{12}^* \leq \pi$. It is easy to verify that $d_{12} \sin\psi$ is a strictly decreasing function when $\theta \in [\theta_{12}^*, \beta]$. This is because $0 < \psi < \frac{\pi}{2}$ and $d_{12} \cos\psi > L_{12}$ for $\theta \in (\theta_{12}^*, \beta)$. Therefore, using the results in [24], we get,

$$\frac{d(d_{12} \sin\psi)}{d\theta} = -d_{12} \cos\psi \left(\frac{d_{12} \cos\psi - L_{12}}{L_{12}}\right) < 0.$$

Similarly, $d_{23} \sin\phi$ reaches a minimum value of 0 when $\theta = 0$ and increases to 2ρ when θ becomes equal to θ_{23}^* (refer to figure 6). Also, $d_{23} \sin\phi$ is a strictly increasing function when $\theta \in [0, \theta_{23}^*]$.

Now, the set $[\theta_{12}^*, \beta] \cap [0, \theta_{23}^*]$ is always nonempty because $0 \leq \beta$ and $\theta_{12}^* \leq \frac{\pi}{2} \leq \theta_{23}^* \leq \pi$. Therefore, one can verify that for $\theta \in [\theta_{12}^*, \beta] \cap [0, \theta_{23}^*]$, the functions $d_{12} \sin\psi$ and $d_{23} \sin\phi$ must intersect at some angle $\theta = \theta^*$. In addition, for any $\theta \in (\theta_{12}^*, \beta) \cap (0, \theta_{23}^*)$,

$$\frac{d^2\mathfrak{D}_1}{d\theta^2} + \frac{d^2\mathfrak{D}_2}{d\theta^2} = d_{12} \cos\psi \left(\frac{d_{12} \cos\psi - L_{12}}{L_{12}}\right) +$$
$$d_{23} \cos\phi \left(\frac{d_{23} \cos\phi - L_{23}}{L_{23}}\right)$$
$$> 0.$$

Hence, $\mathfrak{D}_1(\theta) + \mathfrak{D}_2(\theta)$ reaches an unique minimum at $\theta = \theta^*$. This minimum can be found using an interval bisection algorithm with the number of iterations of the algorithm in the order of $\log_2(\frac{1}{\varepsilon})$ for any given $\varepsilon > 0$ [25]. ∎

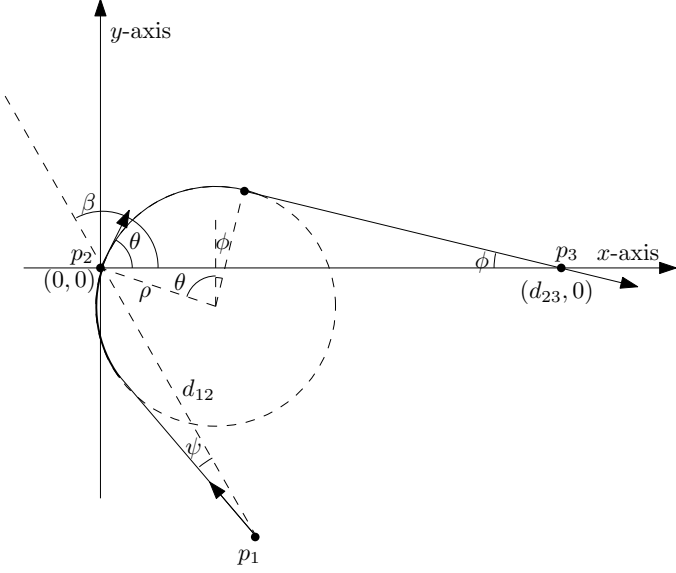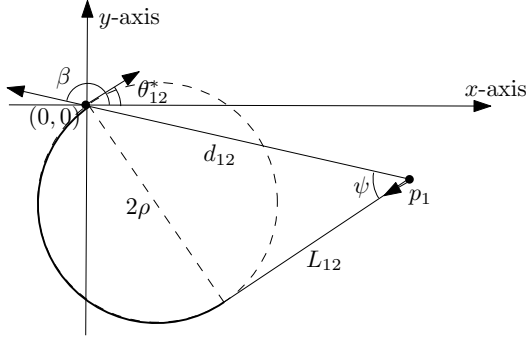Figure 4. A *SRS* path from p_1 to p_3 via p_2.

Figure 5. *SR* path from p_1 to p_2. Note that when $\theta = \theta_{12}^*$, $d_{12} \sin \psi$ reaches a maximum value of 2ρ. As θ is increased further to β, $d_{12} \sin \psi$ reduces to zero.

The following lemma proves an additional property about the nature of the optimal paths for the 3 point problem. This property is useful for numerical implementations and can be used as a termination criteria. Again, the following result is shown for the *SRS* path. *SLS* path can be handled in a similar way.

Lemma 2. *For the optimal SRS path when $y_1 \leq 0$, the angle of turn in the curved segment from point p_1 to p_2 is equal to the angle of turn in the curved segment from p_2 to p_3.*

Proof. Refer to figure 4. Using the result in [24], we get,

$$\frac{d\mathfrak{D}_2}{d\theta} = d_{23} \sin\phi \quad = \rho - \rho\cos(\theta + \phi) = \rho - \rho\cos(turn_{23})$$
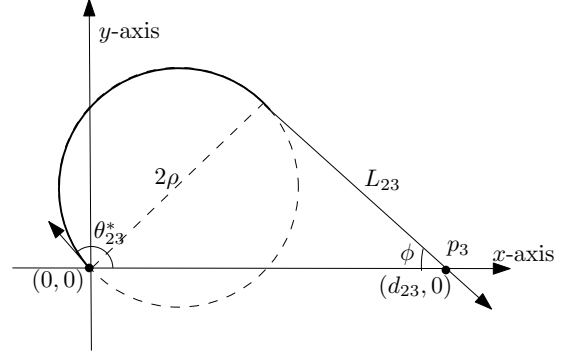

Figure 6. *RS* path from p_2 to p_3. Note that when $\theta = 0$, $d_{12} \sin\phi = 0$. As θ is increased further to θ_{23}^*, $d_{12} \sin\psi$ increases and reaches its maximum value which is equal to 2ρ.

where $turn_{23} = \theta + \phi$ is the turn angle of the curved segment from p_2 to p_3. Similarly,

$$\frac{d\mathfrak{D}_1}{d\theta} = -d_{12} \sin\psi = -(\rho - \rho\cos(turn_{12}))$$

where $turn_{12}$ is the turn angle of the curved segment from p_1 to p_2.

Therefore, $\frac{d\mathfrak{D}_1}{d\theta} + \frac{d\mathfrak{D}_2}{d\theta} = 0$ implies that $\cos(turn_{12}) = \cos(turn_{23})$. Either $turn_{12} = turn_{13}$ or $turn_{12} + turn_{13} = 2\pi$. But $turn_{12} + turn_{13}$ cannot be equal to 2π because this would imply that $y_1 > 0$ which is not true. Therefore, $turn_{12} = turn_{13}$. ∎

4 Approximation algorithm for n points

The approximation algorithm first finds three feasible solutions for the CSP and then chooses the best of these three solutions. The three solutions are constructed in the following way:

1. **Solution $\mathcal{F}_1$:** Use the three point algorithm to find a path for each of the following sequences of points: $(p_1, p_2, p_3), (p_4, p_5, p_6), \cdots, (p_{3k-2}, p_{3k-1}, p_{3k})$. These paths will fix the heading angle of the vehicle at each of the points. Now, use these heading angles to find the shortest Dubins path from p_3 to p_4, p_6 to p_7 and so on. Concatenate all these Dubins paths along with the paths obtained using the three point algorithm such that the resulting solution ($\mathcal{F}_1$) visits the points in the given sequence.

2. **Solution $\mathcal{F}_2$:** Use the three point algorithm to find a path for each of the following sequences of points: $(p_2, p_3, p_4), (p_5, p_6, p_7), \cdots, (p_{3k-4}, p_{3k-3}, p_{3k-2})$. In addition, join points p_{3k-1} and p_{3k} using a line segment. These paths will fix the heading angle of the vehicle at each of the points except p_1. Choose any arbitrary angle for visiting p_1. Now, use these heading angles to find the shortest Dubins path from p_1 to p_2, p_4 to p_5 and so on. Concatenate all these

Dubins paths along with the paths obtained using the three point algorithm and the line segment such that the resulting solution ($\mathcal{F}_2$) visits the points in the given sequence.

3. **Solution** $\mathcal{F}_3$: Use the three point algorithm to find a path for each of the following sequences of points: $(p_3, p_4, p_5), (p_6, p_7, p_8), \cdots, (p_{3k-3}, p_{3k-2}, p_{3k-1})$. In addition, join points p_1 and p_2 using a line segment. These paths will fix the heading angle of the vehicle at each of the points except p_{3k}. Choose any arbitrary angle for visiting p_{3k}. Now, use these heading angles to find the shortest Dubins path from p_2 to p_3, p_5 to p_6 and so on. Concatenate all these Dubins paths along with the paths obtained using the three point algorithm such that the resulting solution ($\mathcal{F}_3$) visits the points in the given sequence.

Among these three solutions, the approximation algorithm chooses a solution ($\mathcal{F}_a$) that corresponds to the least of the lengths of $\mathcal{F}_1$, $\mathcal{F}_2$ and $\mathcal{F}_3$. Let $cost(\mathcal{F})$ denote the length of any solution $\mathcal{F}$. The algorithm runs in polynomial time as it primarily relies on solving the three point problems which can be solved in polynomial time for given $\varepsilon > 0$. The following theorem shows the approximation factor of the algorithm:

Theorem 2. *Let $\mathcal{F}_{opt}$ and $\mathcal{F}_a$ respectively denote an optimal solution to the CSP and the solution obtained by the approximation algorithm. Consider any constant $\varepsilon > 0$. The length of the solution $\mathcal{F}_a$ is at most equal to $(1 + \frac{\pi}{3} + \varepsilon)$ times the length of $\mathcal{F}_{opt}$. That is, $cost(\mathcal{F}_a) \leq (1 + \frac{\pi}{3} + \varepsilon)cost(\mathcal{F}_{opt})$.*

Proof. For $i, j = 1, \cdots, n$, $i < j$, let $\mathcal{F}(p_i, p_j)$ denote the part of the solution $\mathcal{F}$ from point p_i to point p_j. Hence, $cost(\mathcal{F}(p_i, p_j))$ essentially denotes the length of the part of the solution $\mathcal{F}$ from p_i to p_j. Also, let $d_{i,j}$ represent the Euclidean distance between points p_i and p_j.

$$Cost(\mathcal{F}_1) = \sum_{i=1}^{k} cost(\mathcal{F}_1(p_{3i-2}, p_{3i})) + \sum_{i=1}^{k-1} cost(\mathcal{F}_1(p_{3i}, p_{3i+1})). \tag{1}$$

By construction, for any $i = 1, \cdots, k$, the three point algorithm finds a path of bounded curvature with an approximation guarantee of $(1 + \varepsilon)$ from p_{3i-2} to p_{3i}. Therefore, the length of this path must be at most equal to $(1 + \varepsilon)$ times length of the part of the path $\mathcal{F}_{opt}$ from p_{3i-2} to p_{3i}. Hence,

$$cost(\mathcal{F}_1(p_{3i-2}, p_{3i})) \leq (1 + \varepsilon)cost(\mathcal{F}_{opt}(p_{3i-2}, p_{3i})). \tag{2}$$

Similarly, a shortest Dubins path is constructed from p_{3i} to p_{3i+1} for all $i = 1, \cdots, k - 1$. Using the bound shown in [22], we get,

$$cost(\mathcal{F}_1(p_{3i}, p_{3i+1})) \leq (1 + \pi)d_{3i, 3i+1}$$
$$\leq (1 + \pi)cost(\mathcal{F}_{opt}(p_{3i}, p_{3i+1})). \tag{3}$$

Substituting for the bounds from equations (2) and (3) in (1), we get,

$cost(\mathcal{F}_1)$

$$\leq (1 + \varepsilon)\sum_{i=1}^{k} cost(\mathcal{F}_{opt}(p_{3i-2}, p_{3i})) + (1 + \pi)\sum_{i=1}^{k-1} cost(\mathcal{F}_{opt}(p_{3i}, p_{3i+1}))$$

$$= (1 + \varepsilon)\underbrace{\sum_{i=1}^{k} cost(\mathcal{F}_{opt}(p_{3i-2}, p_{3i})) + \sum_{i=1}^{k-1} cost(\mathcal{F}_{opt}(p_{3i}, p_{3i+1}))}_{\leq (1+\varepsilon)cost(\mathcal{F}_{opt})}$$

$$+ \pi \sum_{i=1}^{k-1} cost(\mathcal{F}_{opt}(p_{3i}, p_{3i+1}))$$

$$= (1 + \varepsilon)cost(\mathcal{F}_{opt}) + \pi \sum_{i=1}^{k-1} cost(\mathcal{F}_{opt}(p_{3i}, p_{3i+1})). \tag{4}$$

Similarly, we can also bound the length of solutions $\mathcal{F}_2$ and $\mathcal{F}_3$ as follows:

$$cost(\mathcal{F}_2) \leq (1 + \varepsilon)cost(\mathcal{F}_{opt}) + \pi \sum_{i=1}^{k} cost(\mathcal{F}_{opt}(p_{3i-2}, p_{3i-1})). \tag{5}$$

$$cost(\mathcal{F}_3) \leq (1 + \varepsilon)cost(\mathcal{F}_{opt}) + \pi \sum_{i=1}^{k} cost(\mathcal{F}_{opt}(p_{3i-1}, p_{3i})). \tag{6}$$

We use the following notations to simplify the terms that appear in equations (4),(5) and (6). Let,

$$L_1 = \sum_{i=1}^{k-1} cost(\mathcal{F}_{opt}(p_{3i}, p_{3i+1})).$$

$$L_2 = \sum_{i=1}^{k} cost(\mathcal{F}_{opt}(p_{3i-2}, p_{3i-1})).$$

$$L_3 = \sum_{i=1}^{k} cost(\mathcal{F}_{opt}(p_{3i-1}, p_{3i})).$$

One can verify that $L_1 + L_2 + L_3 = cost(\mathcal{F}_{opt})$. Therefore, as $L_1, L_2, L_3 \geq 0$, we obtain,

$$\min(L_1, L_2, L_3) \leq \frac{1}{3}cost(\mathcal{F}_{opt}). \tag{7}$$

Combining all the bounds for the length of the three solutions and the above equation, we get,

$$cost(\mathcal{F}_a) = \min\left(cost(\mathcal{F}_1), cost(\mathcal{F}_2), cost(\mathcal{F}_3)\right)$$
$$\leq (1+\varepsilon)cost(\mathcal{F}_{opt}) + \pi\min(L_1, L_2, L_3)$$
$$\leq (1 + \frac{\pi}{3} + \varepsilon)cost(\mathcal{F}_{opt}).$$

■

5 Numerical Results

The approximation algorithm was implemented on problem instances with 12, 15, 18, 21, 24, 27 and 30 target points. For a given number of target points, we generated 100 instances. The turning radius of the vehicle was set to 100 units. The coordinates of the points were uniformly randomly generated on a 2D plane with the requirement that the minimum distance between any two adjacent points in the sequence is at least twice the turning radius of the vehicle.

For a given problem instance I, the bound on the *a posterior guarantee* provided by the approximation algorithm is defined as $\frac{C^I_{approx}}{C^I_{lb}}$ where C^I_{approx} is the cost of the feasible solution found by the approximation algorithm and C^I_{lb} is the lower bound on the length of the shortest path for the CSP. The parameter ε in the approximation algorithm was set to 10^{-4}. The lower bound for each instance was obtained using the procedure outlined in [24]. This lower bounding algorithm involved discretizing the set of possible heading angles $[0, 2\pi]$ at each point into 32 uniform intervals (refer to [24] for more details on this computation). All the algorithms were coded in MATLAB and the computations were run on a Dell Precision Workstation (Intel Xeon Processor @2.53 GHz, 12 GB RAM). The average running time of the approximation algorithm was in the order of a second for all the instances.

The max and average *a posterior guarantee* of the solutions found by the approximation algorithm is shown in table 1 along with the theoretical approximation guarantee. Even though the theoretical guarantee is 2.04, the numerical results show that the max and average *a posterior guarantee* was less than 1.28 and 1.15 respectively for the considered instances. These results imply that the proposed algorithm produced solutions with bounds that are significantly better than the guarantees indicated by the approximation factor. The solutions obtained by the approximation algorithm for an instance is shown in figure 7.
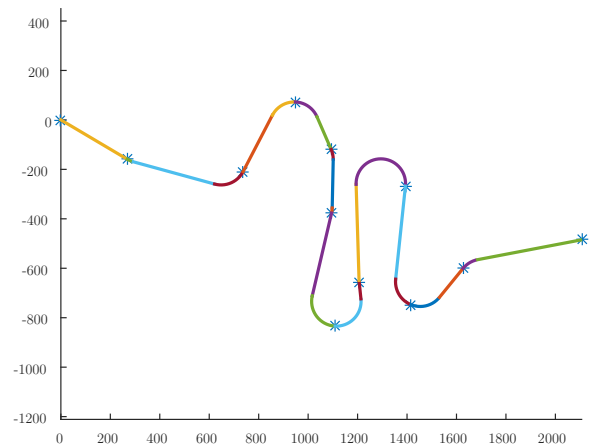
6 Conclusions

This article provided a new approximation algorithm for a curvature constrained shortest path problem (CSP) for visiting a given sequence of target points. First, an algorithm that finds a
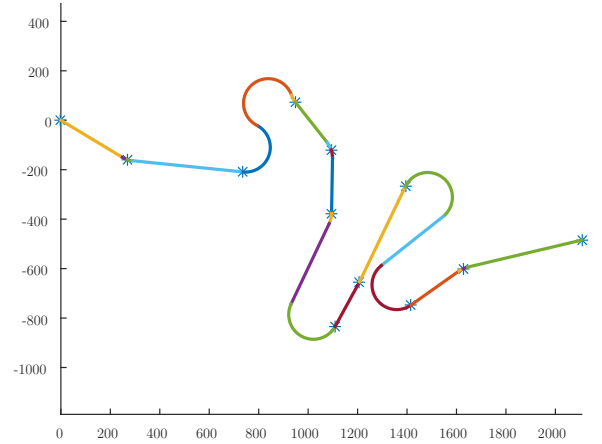
(a) $\mathcal{F}_1$
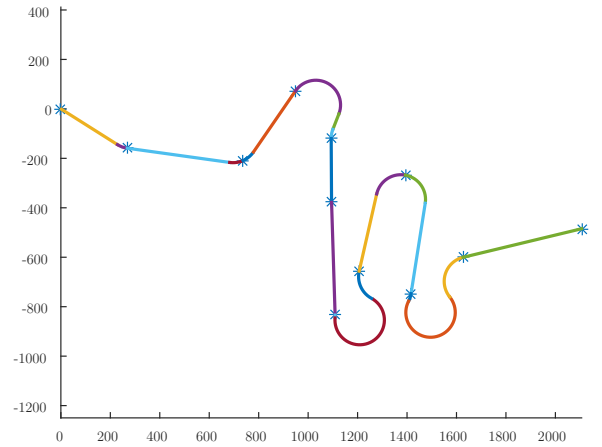
(b) $\mathcal{F}_2$

(c) $\mathcal{F}_3$

Figure 7. The three solutions obtained by the algorithm for a problem instance with 12 points.

Table 1. Comparison of the theoretical and numerical results for the approximation algorithm

No. of Points	Theoretical upper bound	Max a posteriori bound	Average a posteriori bound
12	2.04	1.27	1.09
15	2.04	1.25	1.09
18	2.04	1.24	1.11
21	2.04	1.22	1.11
24	2.04	1.24	1.11
27	2.04	1.23	1.13
30	2.04	1.20	1.12

good path for 3 points is presented. Next, the given sequence is broken into subsequences with 3 points in each subsequence, and a solution is obtained for each subsequence using the three point algorithm. Finally, the solutions corresponding to all the subsequences are concatenated in a suitable way to find a feasible solution to the SPP. This basic idea can be improved in several directions to provide better approximation guarantees for the CSP. First, if one can solve a four point problem with an approximation factor of $(1 + \varepsilon)$ for some small $\varepsilon > 0$, using the ideas presented in this article, a guarantee of $(1 + \frac{\pi}{4} + \varepsilon)$ can be obtained for the CSP. Second, the bounds presented in this article are mainly based on the Euclidean distances with no consideration given to the angle of turn of the vehicle as it passes the points. Including the turn angles has a potential to reduce the approximation factor further. In fact, the algorithm provided in [21] uses the constraints on the turn angles in a clever way to provide a constant factor approximation guarantee. We believe that a combination of the ideas presented in this article with lower bounds that are computed based on the Euclidean distances and the turn angles can improve the approximation guarantees significantly even for the more general case when adjacent points are closed spaced.

REFERENCES

[1] Geiser, K., Slack, D., Allred, E., and Stange, K., 1982. "Irrigation scheduling using crop canopy air temperature differences". *Transactions of the American Society of Agricultural and Biological Engineers,* **25**(3), pp. 689–694.

[2] Jackson, R., Idso, S., Reginato, R., and Pinter, P., 1981. "Canopy temperature as a crop water stress indicator". *Water Resources Research,* **17**(4), pp. 1133–1138.

[3] Ferreira, H., Almeida, C., Martins, A., Almeida, J., Dias, N., Dias, A., and Silva, E., 2009. "Autonomous bathymetry for risk assessment with roaz robotic surface vehicle". In OCEANS 2009 - EUROPE, pp. 1–6.

[4] Casbeer, D., Beard, R., McLain, T., Li, S.-M., and Mehra, R., 2005. "Forest fire monitoring with multiple small uavs". In American Control Conference, 2005. Proceedings of the 2005, pp. 3530–3535 vol. 5.

[5] Corrigan, C. E., Roberts, G. C., Ramana, M. V., Kim, D., and Ramanathan, V., 2008. "Capturing vertical profiles of aerosols and black carbon over the indian ocean using autonomous unmanned aerial vehicles". *Atmospheric Chemistry and Physics,* **8**(3), pp. 737–747.

[6] Zajkowski, T., Dunagan, S., and Eilers, J., 2006. "Small UAS communications mission".

[7] Maza, I., Caballero, F., Capitan, J., de Dios, J. R. M., and Ollero, A., July 2010. "Firemen Monitoring with Multiple UAVs for Search and Rescue Missions".

[8] Krishnamoorthy, K., Casbeer, D., Chandler, P., Pachter, M., and Darbha, S., 2012. "UAV search and capture of a moving ground target under delayed information". In Conference on Decision and Control, IEEE.

[9] Maza, I., Caballero, F., Capitan, J., de Dios, J. R. M., and Ollero, A., 2011. "Experimental Results in Multi-UAV Coordination for Disaster Management and Civil Security Applications". *Journal of Intelligent and Robotic Systems,* **61**, pp. 563–585.

[10] Medeiros, A. C., and Urrutia, S., 2010. "Discrete optimization methods to determine trajectories for dubins' vehicles". *Electronic Notes in Discrete Mathematics,* **36**, pp. 17 – 24. {ISCO} 2010 - International Symposium on Combinatorial Optimization.

[11] Macharet, D., and Campos, M., 2014. "An orientation assignment heuristic to the dubins traveling salesman problem". In *Advances in Artificial Intelligence – IBERAMIA 2014*, A. L. Bazzan and K. Pichara, eds., Vol. 8864 of *Lecture Notes in Computer Science*. Springer International Publishing, pp. 457–468.

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

[13] Macharet, D. G., Neto, A. A., da Camara Neto, V. F., and Campos, M. F., 2012. "Data gathering tour optimization for dubins' vehicles". In Congress on Evolutionary Computation (CEC), pp. 1–8.

[14] Sujit, P., Hudzietz, B., and Saripalli, S., 2013. "Route planning for angle constrained terrain mapping using an unmanned aerial vehicle". *Journal of Intelligent & Robotic Systems,* **69**(1-4), pp. 273–283.

[15] Kenefic, R. J., 2008. "Finding good dubins tours for uavs using particle swarm optimization". *Journal of Aerospace Computing, Information, and Communication,* **5**(2), pp. 47–56.

[16] Macharet, D. G., Neto, A. A., da Camara Neto, V. F., and Campos, M. F., 2011. "Nonholonomic path planning optimization for dubins' vehicles". In IEEE International Conference on Robotics and Automation (talk), pp. 4208–4213.

[17] Tang, Z., and Ozguner, U., 2005. "Motion planning for multitarget surveillance with mobile sensor agents". *IEEE Transactions on Robotics,* **21**(5), Oct, pp. 898–908.

[18] Ma, X., Castañón, D., et al., 2006. "Receding horizon planning for dubins traveling salesman problems". In Conference on Decision and Control, IEEE, pp. 5453–5458.

[19] Rathinam, S., Sengupta, R., and Darbha, S., 2007. "A resource allocation algorithm for multivehicle systems with nonholonomic constraints". *IEEE Transactions on Automation Science and Engineering,* **4**, pp. 98–104.

[20] L.E.Dubins, 1957. "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents". *American Journal of Mathematics,* **79**(3), pp. 487–516.

[21] Lee, J.-H., Cheong, O., Kwon, W.-C., Shin, S., and Chwa, K.-Y., 2000. "Approximation of curvature-constrained shortest paths through a sequence of points". In *Algorithms - ESA 2000*, M. Paterson, ed., Vol. 1879 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, pp. 314–325.

[22] Kim, H.-S., and Cheong, O., 2013. "The cost of bounded curvature". *Computational Geometry,* **46**(6), pp. 648 – 672.

[23] Boissonnat, J.-D., and Bui, X.-N., 1994. Accessibility region for a car that only moves forwards along optimal paths. Tech. Rep. RR-2181, INRIA.

[24] Manyam, S., and Rathinam, S., 2015. "A tight lower bounding procedure for the dubins traveling salesman problem". *Presented at the International Symposium on Mathematical Programming*.

[25] Nemirovski, A., 1994. Efficient methods in convex programming, lecture notes.