Algorithms for the Traveling Salesman Problem with Neighborhoods Involving a Dubins Vehicle

Jason T. Isaacs, Daniel J. Klein, João P. Hespanha

***Abstract*— We study the problem of finding the minimum-length curvature constrained closed path through a set of regions in the plane. This problem is referred to as the Dubins Traveling Salesperson Problem with Neighborhoods (DTSPN). Two algorithms are presented that transform this infinite dimensional combinatorial optimization problem into a finite dimensional asymmetric TSP by sampling and applying the appropriate transformations, thus allowing the use of existing approximation algorithms. We show for the case of disjoint regions, the first algorithm needs only to sample each region once to produce a tour within a factor of the length of the optimal tour that is independent of the number of regions. We present a second algorithm that performs no worse than the best existing algorithm and can perform significantly better when the regions overlap.**

I. Introduction

Research in the area of unmanned aerial vehicles (UAV) has evolved in recent years. There is rich literature covering various areas of autonomy including path planning, trajectory planning, task allocation, cooperation, sensing, and communications. As the mission objectives of UAVs have increased in complexity and importance, problems are starting to arise at the intersection of these disciplines. The Dubins Traveling Salesman Problem with Neighborhoods (DTSPN) combines the problem of path planning with trajectory planning while using neighborhoods to represent communication ranges or sensor footprints. In this problem the UAV simply needs to enter a region surrounding each objective waypoint.

The path planning problem seeks to determine the optimal sequence of waypoints to visit in order to meet certain mission objectives while minimizing costs, such as the total length of the mission [1], [2]. Path planning problems typically rely on approximating the cost of the mission by the length of the solution to an Euclidean Traveling Salesman Problem (ETSP), where the cost to travel from one waypoint to the next is approximated by the Euclidean distance between the two waypoints. This approximation simplifies the overall optimization, but may lead to UAV routes that are far from optimal because the aircraft dynamics are not considered.

Another area of UAV research is trajectory planning, in which the goal given an initial and final waypoint pair is to determine the optimal control inputs to reach the final waypoint in minimum time given dynamic constraints of the aircraft. In 1957, Dubins showed that for an approximate model of aircraft dynamics the optimal motion between a pair of waypoints can be chosen among six possible paths [3]. Similar results were proven later in [4] using tools from optimal control theory. In [5], the authors propose a means of choosing the optimal Dubins path without computing all six possible Dubins optimal paths.

A significant amount of research has gone into combining the problems of motion planning and path planning [6], [7], [8], [9], [10], and [11]. In these works, the dynamics of the UAV are taken into consideration by using the Dubins model when determining the optimal sequence of waypoints. This problem is typically referred to as the Dubins Traveling Salesman Problem (DTSP).

A third area of UAV related research is a version of path planning that takes into account the communication range of the aircraft or the sensor footprint of the aircraft. This problem is best described as a Traveling Salesman Problem with Neighborhoods (TSPN). Now, not only does one determine a sequence of regions but also an entry point at each region. Many researchers have addressed this problem with various regions, but most have used the Euclidean distance as the cost function [12], [13], [14]. Obermeyer was the first to tackle the TSPN with Dubins vehicle dynamics in [15] using a genetic algorithm approach, then later in [16] by using a sampling based roadmap method which we will call RCM that is proven to be resolution complete. In the latter method, the DTSPN is transformed to a General Traveling Salesman Problem (GTSP) with non-overlapping nodesets and then to an Asymmetric Traveling Salesmen Problem (ATSP) through a version of the Noon and Bean transformation [17]. This is a similar approach to that used in [9] for the DTSP.

In this work, we propose two algorithms to approximate the DTSPN. The first involves the case in which the regions do not overlap. In this case, we present a simple algorithm requiring just one sample configuration in each region. The second algorithm addresses the case in which the regions intersect frequently. In this case, we propose a sampling based roadmap algorithm similar to that of [16], but use a more general version of the Noon and Bean transformation [18] in which the GTSP can contain overlapping nodesets. We show that for the same set of samples this method will produce a tour that is no longer than that of RCM from [16] and performs significantly better when the regions overlap.

The remainder of the paper is organized as follows. In Section II, the Dubins Traveling Salesman Problem with

This work was supported by the Institute for Collaborative Biotechnologies through contract No. W911NF-09-D-0001 from the U.S. Army Research Office.

Authors are with the Center for Control, Dynamical Systems, and Computation, and the Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106 USA. {jtisaacs,djklein,hespanha}@ece.ucsb.edu

Neighborhoods is formally introduced. An approximation algorithm for the DTSPN when the regions do not overlap is described in Section III. Section IV describes a second algorithm for the DTSP when the regions overlap. In Section V, we present a numerical study comparing our algorithm with an existing algorithm for various sized regions and various amounts of overlap. Conclusions and future work are discussed in Section VI.

II. Problem Statement

The dynamics of the UAV can be approximated by the Dubins vehicle in the plane. The state of the Dubins vehicle $\mathbf{X}$ can be represented by the triplet $(x, y, \theta) \in SE(2)$, where $(x, y) \in \mathbb{R}^2$ define the position of the vehicle in the plane and $\theta \in \mathbb{S}^1$ defines the heading of the vehicle. The vehicle dynamics are then written as,

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \nu\cos(\theta) \\ \nu\sin(\theta) \\ \frac{\nu}{\rho}u \end{bmatrix}, \tag{1}$$

where ν is the forward speed of the vehicle, ρ is the minimum turning radius, and $u \in [-1, 1]$ is the bounded control input. Let $\mathcal{C}_\rho : SE(2) \times SE(2) \to \mathbb{R}_+$ associate the length $\mathcal{C}_\rho(\mathbf{X}_1, \mathbf{X}_2)$ of the minimum length path from an initial configuration $\mathbf{X}_1$ of the Dubins vehicle to a final configuration $\mathbf{X}_2$, subject to the dynamic constraints in (1). This length, which we will refer to as the Dubins distance from $\mathbf{X}_1$ to $\mathbf{X}_2$, can be computed in constant time [5].

Let $\mathcal{R} = \{\mathcal{R}_1, \mathcal{R}_2, ..., \mathcal{R}_n\}$ be set of n compact regions in a compact region $\mathcal{Q} \subset \mathbb{R}^2$, and let $\mathbf{\Sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_n)$ be an ordered permutation of $\{1, \ldots, n\}$. Define a projection from $SE(2)$ to $\mathbb{R}^2$ as $\mathcal{P} : SE(2) \to \mathbb{R}^2$, i.e. $\mathcal{P}(\mathbf{X}) = \begin{bmatrix} x & y \end{bmatrix}^T$, and let $\mathbf{P}_i$ be a point in $SE(2)$ whose projection lies in $\mathcal{R}_i$. We denote the vector created by stacking all n configurations $\mathbf{P}_i$ as $\mathbf{P} \in SE(2)^n$.

The DTSPN involves finding the minimum length tour in which the Dubins vehicle visits each region in $\mathcal{R}$ while obeying the kinematic constraints of (1). This is an optimization over all possible permutations $\mathbf{\Sigma}$ and configurations $\mathbf{P}$. Stated more formally:

Problem 2.1 (DTSPN):

$$\begin{aligned} \underset{\mathbf{\Sigma},\mathbf{P}}{\text{minimize}} \quad & \mathcal{C}_\rho(\mathbf{P}_{\sigma_n}, \mathbf{P}_{\sigma_1}) + \sum_{i=1}^{n-1} \mathcal{C}_\rho(\mathbf{P}_{\sigma_i}, \mathbf{P}_{\sigma_{i+1}}) \\ \text{subject to} \quad & \mathcal{P}(\mathbf{P}_i) \in \mathcal{R}_i,\ i = 1, \ldots, n. \end{aligned}$$

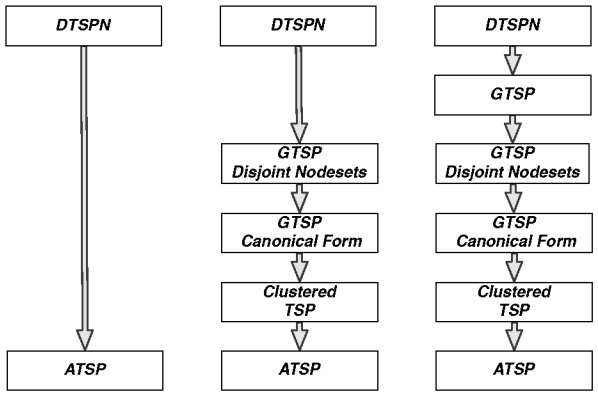
We present two algorithms to address this problem. Each algorithm involves generating a set of $m \geq n$ sample configurations $\mathbf{S} := \{\mathbf{S}_1, \ldots, \mathbf{S}_m\}$ such that,

$$\mathcal{P}(\mathbf{S}_k) \in \bigcup_{i=1}^{n} \mathcal{R}_i, \tag{2}$$

$$\forall i\ \exists k \text{ s.t. } \mathcal{P}(\mathbf{S}_k) \in \mathcal{R}_i. \tag{3}$$

Each algorithm approximates Problem 2.1 by finding the best sample configurations $\mathbf{P} \subseteq \mathbf{S}$ and the order $\mathbf{\Sigma}$ in which to visit them.

III. DTSPN with Disjoint Regions



(a) Algorithm 1. (b) RCM [16]. (c) Algorithm 2.

Fig. 1. Three algorithms for DTSPN of increasing generality: (a) disjoint regions and disjoint nodesets, (b) overlapping regions and disjoint nodesets and (c) overlapping regions and overlapping nodesets.

Consider the instance of Problem 2.1 where all the regions are disjoint. In this simpler case, we consider an algorithm that finds a feasible solution with a length that is within a factor of the length of the optimal solution that is independent of n. This algorithm samples from each region a single Dubins configuration, computes the $n(n-1)$ Dubins distances between those configurations, and uses these costs to form a cost matrix $\mathbf{D}$ used by the ATSP solver, see Algorithm 1. The process of taking a single sample per region allows one to proceed directly to the ATSP step, as can be seen in Figure 1(a).

Algorithm 1 DTSPN$(\rho, \mathcal{R})$

Require: An asymmetric TSP aproximation algorithm ATSP($\mathbf{D}$), and a function to compute the optimal Dubins distance between two configurations $\mathcal{C}_\rho(\mathbf{X}, \mathbf{X}')$.

Ensure: A sequence $\hat{\mathbf{\Sigma}}$ of waypoint configurations $\hat{\mathbf{P}}$ for DTSPN over the set of disjoint regions $\mathcal{R}$.

```
for i = 1 to n do
    set S_i := {X ∈ SE(2) | P(X) ∈ R_i}.
end for
for i = 1 to n do
    for j = 1 to n do
        set D(i, j) := C_ρ(S_i, S_j).
    end for
end for
set Σ̂ :=ATSP(D), P̂ := S
return Σ̂, P̂
```

Define ϵ and μ as the minimum and maximum Euclidean distances between any two regions in $\mathcal{R}$,

$$\epsilon := \min_{i \neq j} \min_{(\mathbf{X}, \mathbf{X}') \in \mathcal{R}_i \times \mathcal{R}_j} \|\mathcal{P}(\mathbf{X}) - \mathcal{P}(\mathbf{X}')\|_2, \tag{4}$$

$$\mu := \max_{i \neq j} \min_{(\mathbf{X}, \mathbf{X}') \in \mathcal{R}_i \times \mathcal{R}_j} \|\mathcal{P}(\mathbf{X}) - \mathcal{P}(\mathbf{X}')\|_2, \tag{5}$$

respectively. Define the diameter d_i of region $\mathcal{R}_i$, as the maximum Euclidean distance between any two points in the region $\mathcal{R}_i$,

$$d_i := \max_{(\mathbf{X},\mathbf{X}')\in\mathcal{R}_i\times\mathcal{R}_i} \|\mathcal{P}(\mathbf{X}) - \mathcal{P}(\mathbf{X}')\|_2, \tag{6}$$

and define d as the maximum diameter of all the regions,

$$d := \max_{i\in\{1,2,\ldots,n\}} d_i. \tag{7}$$

Theorem 3.1 (Worst case performance for Algorithm 1): Given $\rho > 0$, and $n \geq 2$ disjoint regions $\mathcal{R}$ whose maximum diameter is d, and separated by a least $\epsilon > 0$, Algorithm 1 will produce a DTSPN tour that is within the following factor of the optimum DTSPN tour. If an exact ATSP solver is used, the factor will be,

$$\lambda := \left(\frac{2d + \mu + \kappa\rho\pi}{\epsilon}\right). \tag{8}$$

If instead the modified Christofides algorithm from [19] is used, the factor will be

$$\lambda \min\left(\log n, \frac{3}{2}\gamma\right) \tag{9}$$

where

$$\gamma := \left(1 + \frac{\kappa\rho\pi}{\epsilon}\right), \tag{10}$$

comes from [8]

The proof of Theorem 3.1 will make use of Theorem 3.4 from [7], which bounds the Dubins distance as,

$$\mathcal{C}_\rho(\mathbf{P}, \mathbf{P}') \leq \|\mathcal{P}(\mathbf{P}) - \mathcal{P}(\mathbf{P}')\|_2 + \kappa\pi\rho, \tag{11}$$

where $\kappa \in [2.657, 2.658]$.

Proof: The Euclidean distance between any two configurations $\mathbf{P}_i$ and $\mathbf{P}_j$ is at most $2d + \mu$. Using (11),

$$\mathcal{C}_\rho(\mathbf{P}_i, \mathbf{P}_j) \leq 2d + \mu + \kappa\pi\rho. \tag{12}$$

Let $\mathbf{P}^*$ and $\mathbf{\Sigma}^*$ denote the optimal tour. Then, the length of the each leg of the tour provided by Algorithm 1 satisfies

$$\frac{\mathcal{C}_\rho(\hat{\mathbf{P}}_{\hat{\sigma}_i}, \hat{\mathbf{P}}_{\hat{\sigma}_{i+1}})}{\mathcal{C}_\rho(\mathbf{P}^*_{\sigma^*_i}, \mathbf{P}^*_{\sigma^*_{i+1}})} \leq \left(\frac{2d + \mu + \kappa\rho\pi}{\epsilon}\right) = \lambda, \tag{13}$$

because $\mathcal{C}_\rho(\mathbf{P}^*_{\sigma^*_i}, \mathbf{P}^*_{\sigma^*_{i+1}}) \geq \epsilon$.

The total tour length for a given ρ, $\mathcal{R}$, $\mathbf{\Sigma}$, and $\mathbf{P}$ is computed by summing Dubins distances along the tour,

$$\mathcal{L}_{\rho,\mathcal{R}}(\mathbf{\Sigma}, \mathbf{P}) := \mathcal{C}_\rho(\mathbf{P}_{\sigma_n}, \mathbf{P}_{\sigma_1}) + \sum_{i=1}^{n-1} \mathcal{C}_\rho(\mathbf{P}_{\sigma_i}, \mathbf{P}_{\sigma_{i+1}}), \tag{14}$$

so the length of the tour produced by Algorithm 1 satisfies

$$\mathcal{L}_{\rho,\mathcal{R}}(\hat{\mathbf{\Sigma}}, \hat{\mathbf{P}}) \leq \lambda \mathcal{L}_{\rho,\mathcal{R}}(\mathbf{\Sigma}^*, \mathbf{P}^*). \tag{15}$$

The above result holds if an exact TSP solve is used, and matches the condition given in (8). If using an approximate ATSP solver, the bound will be slightly weaker. For example Christofides' algorithm has a multiplicative error bound of $\min\left(\log n, \frac{3}{2}\gamma\right)$, resulting in an approximate DTSPN solution with the combined approximation factor given in (9). ■

Given n samples from n regions this algorithm will compute the ATSP over n nodes. The worst case computational complexity of filling the ATSP cost matrix is upper bounded by $O(n^2)$. Then the worst case complexity for solving the ATSP using the modified version of Christofides' algorithm provided in [19] is upper bounded by $O(n^3)$.

IV. DTSPN WITH INTERSECTING REGIONS

A. Algorithm 2

For the case in which the regions overlap, we propose a more general algorithm that seeks to take advantage of the possibility of visiting several regions at a single vehicle configuration. Algorithm 2 approximates the solution to Problem 2.1 by converting the DTSPN to a GTSP with overlapping nodesets, as shown in Figure 1(c), by sampling regions $\mathcal{R}$ with a finite set of m Dubins vehicle configurations $\mathbf{S}$. The GTSP is then transformed into a standard ATSP through the Noon and Bean transformation [18]. A variety of solvers are available for ATSP.

The GTSP can be described with a directed graph with nodes $\mathcal{N}$ and arcs $\mathcal{A}$ where the nodes are members of predefined nodesets $\mathcal{S}$. Here each node represents an element of the vector of sampled configurations $\mathbf{S}$, and the arc connecting node $\mathbf{S}_i$ to node $\mathbf{S}_j$ represents the length of the minimum length path for a Dubins vehicle $c_{i,j} = \mathcal{C}_\rho(\mathbf{S}_i, \mathbf{S}_j)$ from configuration $\mathbf{S}_i$ to configuration $\mathbf{S}_j$. The nodeset $\mathcal{S}_k$ corresponding to region $\mathcal{R}_k$ contains all samples whose projection lies in $\mathcal{R}_k$, $\mathcal{S}_k := \{\mathbf{S}_i \mid \mathcal{P}(\mathbf{S}_i) \in \mathcal{R}_k\}$ for $i \in \{1, 2, \ldots, m\}$. The objective of the GTSP is to find a minimum cost cycle passing through each nodeset exactly one time.

B. Noon and Bean Transformation

What follows is a brief summary of the Noon-Bean transformation from [18] as it is used in this work. The transformation is best described in three stages.

The first stage converts the GTSP to a GTSP with mutually exclusive nodesets. This is done by first eliminating any arcs from $\mathcal{A}$ that do not enter at least one new nodeset. Next, a finite cost $\alpha \geq \sum_{(i,j)\in\mathcal{A}} c_{i,j}$ is added to each arc cost for each new nodeset the arc enters. Next, any nodes that belong to more than one nodeset are duplicated and placed in different nodesets so as to allow each node to have membership in only one nodeset. Any arcs to and from the original nodes are duplicated as well. In addition, zero cost arcs are added between all the spawned nodes of each multiple membership node. The large cost α added to all the other arcs ensures that all spawned nodes will be visited consecutively, if at all.

The second stage takes the GTSP with mutually exclusive nodesets and eliminates any intraset arcs, leaving a GTSP in "canonical form." The third stage of the transformation converts the canonical GTSP to a "clustered" TSP as follows. The nodes in each nodeset are first enumerated. Then, a

zero cost cycle is created for each nodeset by adding zero cost edges between consecutive nodes in each nodeset and connecting the first node to the last. The *interset* edges are then shifted so they emanate from the previous node in its cycle. Finally, the clustered TSP is converted to an ATSP by adding a finite cost $\beta \geq \sum_{(i,j)\in\mathcal{A}} c_{i,j}$ to each intercluster arc cost.

C. Performance Comparison

Algorithm 2 is very similar to the Resolution Complete Method (RCM) proposed in [16] with the key exception that we use the fact that visiting one of these samples in the intersection of multiple regions achieves the goal of visiting all these regions. The RCM requires mutually exclusive nodesets for the conversion from DTSPN to a GTSP with disjoint nodesets, as depicted in Figure 1(b). To meet this requirement, samples are assigned directly to the nodeset of the region from whose boundary they are drawn. If multiple regions overlap and a sample lies in the intersection, Algorithm 2 assigns this sample to all the nodesets corresponding to all the intersecting regions, while RCM does not. Algorithm 2 then uses this additional information in the optimization.

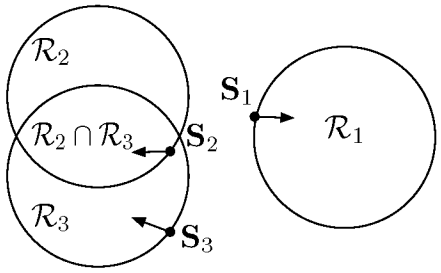
Theorem 4.1 (Algorithm 2 performance): Given $\rho > 0$, $n \geq 2$ possibly intersecting regions $\mathcal{R}$, and m sample configurations $\mathbf{S}$, let T_{A2} and T_{RCM} denote the tours produced by Algorithm 2 and the RCM [16], respectively. Then the length of T_{A2} is no greater than that of T_{RCM},

$$\text{length}(T_{A2}) \leq \text{length}(T_{RCM}). \tag{16}$$

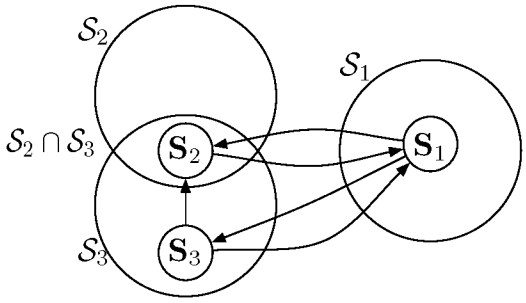
Proof: Let $T = \{\mathbf{S}_1, \mathbf{S}_2, \ldots, \mathbf{S}_n\}$ be a feasible tour, and note that both Algorithm 2 and RCM minimize the tour length plus an additive constant while ensuring that all regions are visited. The difference is that Algorithm 2 may produce tours visiting fewer than n unique samples, should some samples lie in the multiple regions. In particular, Algorithm 2 ensures that each leg of the tour enters at least one new region, by construction. Therefore, in performing the optimization Algorithm 2 will either consider T, or subset of T, in which samples at the end of legs not entering an unvisited region have been removed. Due to the Dubins distance function satisfying the triangle inequality [20], a tour that visits a redundant sample will be longer than a tour that visits a subset of the samples. The optimal tour T_{A2} cannot be longer than T_{RCM}, because both optimize over the same set of feasible tours except for the tours in which Algorithm 2 bypasses these unneeded samples. ■

The property *resolution complete method* as used in [16], dictates that the method converges to a solution at least as good as any nonisolated optimum solution as the number of sample configurations goes to infinity.

Corollary 4.2 (Algorithm 2 is Resolution Complete): Given $\rho > 0$, $n \geq 2$ possibly intersecting regions $\mathcal{R}$, and m sample configurations $\mathbf{S}$ drawn from a Halton quasirandom sequence [21] as in RCM, Algorithm 2 is Resolution Complete.



(a) Example instance of DTSPN with three circular regions and samples $\mathbf{S}_1$, $\mathbf{S}_2$, and $\mathbf{S}_3$.



(b) "GTSP with overlapping nodesets" from Algorithm 2 with edges only entering new nodesets.

Fig. 2. Example DTSPN with the corresponding "GTSP with overlapping nodesets".

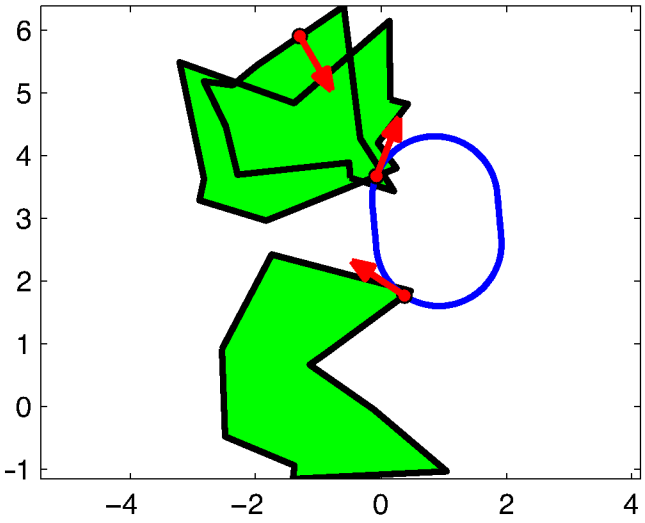
Proof: From [16], the RCM is a resolution complete method and converges as the number of samples goes to infinity, and from Theorem 4.1, we have shown that for the same set of sample configurations Algorithm 2 will produce a tour that is no longer than RCM. ■

D. Complexity of Algorithm 2

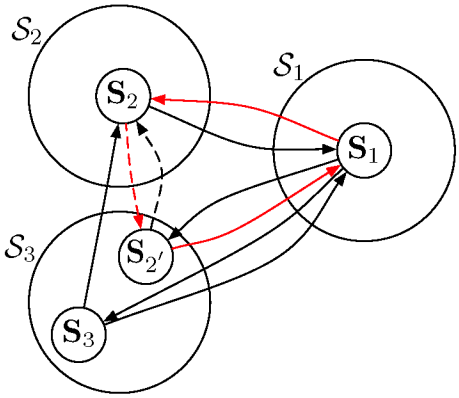
We have provided an algorithm that takes advantage of when sample configurations happen to lie in overlapping regions, and we have shown that this algorithm produces a tour that is no longer than the previous best algorithms in the literature. However, the size of the ATSP is increased by the number of multiple nodeset duplicate nodes. Given a m samples from n regions this algorithm will compute the ATSP over at most mn nodes. The worst case computational complexity of the Noon and Bean transformation [18] is upper bounded by $O(m^2n^4)$. Then the worst case complexity for solving the ATSP using the modified version of Christofides' algorithm provided in [19] is upper bounded by $O(m^3n^3)$.

V. Numerical Results

In Theorem 4.1, we have shown that for the same sample set Algorithm 2 will perform no worse than the resolution complete method from [16], but at the cost of solving a larger ATSP problem when there exist samples that are contained in multiple regions. In this section, we use Monte Carlo Simulation to investigate the level of performance improvement that can be gained by using Algorithm 2.

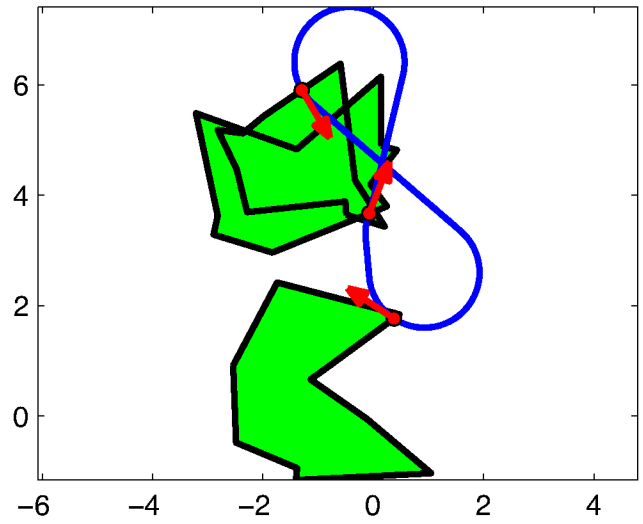


(a) Example Tour: Algorithm 2, Tour Length = 7.7.

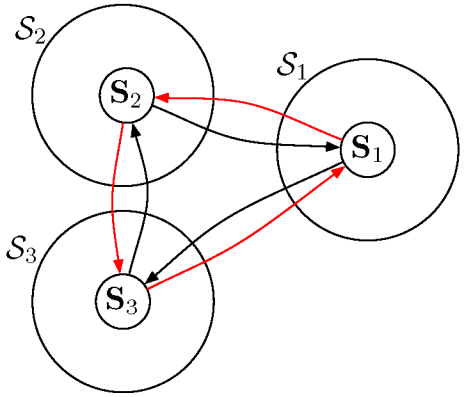


(b) "GTSP with mutually exclusive nodesets" from Algorithm 2 with duplicated node $\mathbf{S}_{2'}$ and zero cost edges between $\mathbf{S}_2$ and $\mathbf{S}_{2'}$.

Fig. 3. Example DTSPN instance with corresponding "GTSP with mutually exclusive" nodesets from Algorithm 2.



(a) Example Tour: RCM, Tour Length = 15.4.



(b) "GTSP with mutually exclusive nodesets" from RCM with no duplicated node $\mathbf{S}_{2'}$ and nonzero cost edge from $\mathbf{S}_2$ to $\mathbf{S}_3$.

Fig. 4. Example DTSPN instance with corresponding "GTSP with mutually exclusive" nodesets from RCM.

The centers of circular regions are variable but homogeneous diameter are randomly placed in a square of variable side length. By varying both the size of regions and the area in which the centers of the regions are confined we are able to vary the degree of overlap. The turning radius of the UAV ρ is set to unit radius. To solve for the tours we used the symmetric TSP solver *linkern* available at [22], which uses the Chained Lin-Kernighan Heuristic from [23]. The radii of the circular regions were varied over $\{0.5, 0.75, 1.00, \ldots, 5.5\}$ and the length of the sides of the square were varied over $\{5, 5.5, \ldots, 15\}$. For the first test, we ran 100 trials where 10 regions were randomly placed in the bounding box and 50 samples were drawn from the boundaries of the regions. In a second test, we repeated the same test parameters with 100 samples drawn from the boundaries of the regions.

The results can be seen in Figures 5 and 6 respectively, where the ratio of the average length of the tours found by the

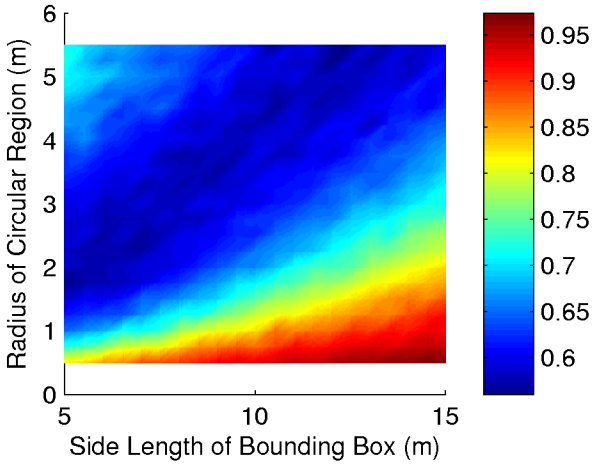


Fig. 5. Simulation results for the 50 sample test.

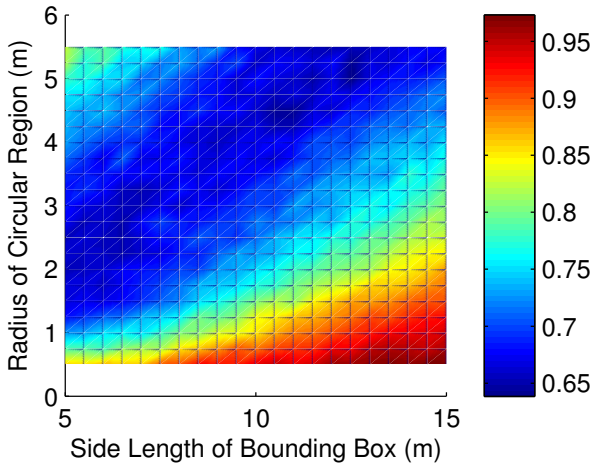


Fig. 6. Simulation results for the 100 sample test.

Algorithm 2 to those found by RCM are displayed for each test configuration. In both instances, it is clear that for small regions and large bounding box (bottom right of plots) that there is little to no overlap and the two algorithms perform equivalently. The tests of interest are when the regions grow and the bounding area shrinks (moving from bottom right to top left). For these cases we see that on average Algorithm 2 finds tours that around nearly half the length of RCM.

VI. Conclusion

The work in this paper has introduced two algorithms addressing the Dubins Traveling Salesman Problem with Neighbhorhoods. When the neighborhoods or regions of interest do not intersect, we have shown that Algorithm 1 achieves an approximation factor independent of the number of regions with a worst case complexity of $O(n^3)$ due to the approximation of the ATSP. The more general case where the regions of interest intersect frequently was addressed in Algorithm 2. This algorithm samples the regions and then relies on the Noon and Bean tranformation [18] for overlapping nodesets to transorm the problem to an ATSP. We show that for the same set of samples this method will produce a tour that is no longer than that of [16], and presented numerical results that show performance improvement when there is overlap in the regions of interest.

There are many directions in which this work may be extended. First, it is of interest to understand if a deterministic way to sample the configurations would be of benefit. For instance, if there is significant overlap would it be beneficial to ensure that at least one sample is taken from each subregion. Finally, we are applying this work to improve the work in [1] to account for communication regions and Dubins dynamics.

References

[1] D. J. Klein, J. Schweikl, J. T. Isaacs, and J. P. Hespanha, "On UAV routing protocols for sparse sensor data exfiltration," in *American Control Conference (ACC)*, June 2010, pp. 6494 –6500.

[2] S. D. Bopardikar, S. L. Smith, F. Bullo, and J. P. Hespanha, "Dynamic vehicle routing for translating demands: Stability analysis and receding-horizon policies," *IEEE Transactions on Automatic Control*, vol. 55, no. 11, 2010.

[3] 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. 497–516, 1957.

[4] J.-D. Boissonnat, A. Cérézo, and J. Leblond., "Shortest paths of bounded curvature in the plane." *Journal of Intelligent and Robotics Systems*, vol. 11, no. 1-2, pp. 5–20, 1994.

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

[6] K. Savla, E. Frazzoli, and F. Bullo, "On the point-to-point and traveling salesperson problems for Dubins' vehicle," in *American Control Conference (ACC)*, 2005, pp. 786 – 791.

[7] ——, "Traveling salesperson problems for the Dubins vehicle," *IEEE Transactions on Automatic Control*, vol. 53, no. 6, pp. 1378 –1391, July 2008.

[8] J. Le Ny, E. Frazzoli, and E. Feron, "The curvature-constrained traveling salesman problem for high point densities," in *IEEE Conference on Decision and Control*, Dec. 2007, pp. 5985 –5990.

[9] J. Le Ny, E. Feron, and E. Frazzoli, "On the curvature-constrained traveling salesman problem," *IEEE Transactions on Automatic Control*, To appear.

[10] X. Ma and D. Castanon, "Receding horizon planning for Dubins traveling salesman problems," Dec. 2006, pp. 5453 –5458.

[11] S. Rathinam, R. Sengupta, and S. Darbha, "A resource allocation algorithm for multiple vehicle systems with non-holnomic constraints," *IEEE Transactions on Automation Science and Engineering*, vol. 4, no. 1, pp. 98–104, Jan. 2007.

[12] A. Dumitrescu and J. S. B. Mitchell, "Approximation algorithms for TSP with neighborhoods in the plane," *Journal of Algorithms*, vol. 48, no. 1, pp. 135 – 159, 2003.

[13] K. Elbassioni, A. V. Fishkin, N. H. Mustafa, and R. Sitters, "Approximation algorithms for Euclidean group TSP," in *32nd International Colloquim Automata, Languages and Programming (ICALP)*. Springer, 2005, pp. 1115–1126.

[14] B. Yuan, M. Orlowska, and S. Sadiq, "On the optimal robot routing problem in wireless sensor networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 9, pp. 1252 –1261, Sept. 2007.

[15] K. J. Obermeyer, "Path planning for a UAV performing reconnaissance of static ground targets in terrain," in *AIAA Conference on Guidance, Navigation, and Control*, Chicago, IL, USA, August 2009.

[16] K. J. Obermeyer, O. P., and D. S., "Sampling-based roadmap methods for a visual reconnaissance UAV," in *AIAA Conference on Guidance, Navigation, and Control*, Toronto, ON, Canada, August 2010.

[17] C. E. Noon and J. C. Bean, "An efficient transformation of the generalized traveling salesman problem," Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, Tech. Rep. 91-26, 1991.

[18] ——, "An efficient transformation of the generalized traveling salesman problem," Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, Tech. Rep. 89-36, 1989.

[19] A. Frieze, G. Galbiati, and F. Maffioli, "On the worst-case performance of some algorithms for the asymmetric traveling salesman problem." *Networks*, vol. 12, pp. 23–39, 1982.

[20] S. Yadlapalli, W. Malik, S. Darbha, and S. Rathinam, "A lagrangian-based algorithm for a combinatorial motion planning problem," in *Advances in Cooperative Control and Optimization*, ser. Lecture Notes in Control and Information Sciences, P. Pardalos, R. Murphey, D. Grundel, and M. Hirsch, Eds. Springer Berlin / Heidelberg, 2007, vol. 369, pp. 373–387.

[21] J. H. Halton, "On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals," *Numerische Mathematik*, vol. 2, pp. 84–90, 1960.

[22] D. Applegate, R. Bixby, V. Chvátal, and W. Cook, "Concorde TSP solver," Website: http//www.tsp.gatech.edu/concorde.

[23] D. Applegate, W. Cook, and A. Rohe, "Chained Lin-Kernighan for large traveling salesman problems," *INFORMS Journal on Computing*, vol. 15, no. 1, pp. 82–92, 2003.