2020 IEEE International Conference on Robotics and Automation (ICRA)
31 May - 31 August, 2020. Paris, France

DC-CAPT: Concurrent Assignment and Planning of Trajectories for Dubins Cars

Michael Whitzer, Daigo Shishika, Dinesh Thakur, Vijay Kumar, Amanda Prorok

Abstract— **We present an algorithm for the concurrent assignment and planning of collision-free trajectories (DC-CAPT) for robots whose kinematics can be modeled as Dubins cars, i.e., robots constrained in terms of their initial orientation and their minimum turning radius. Coupling the assignment and trajectory planning subproblems allows for a computationally tractable solution. This solution is guaranteed to be collision-free through the use of a single constraint: the start and goal locations have separation distance greater than some threshold. We derive this separation distance by extending a prior work that assumed holonomic robots. We demonstrate the validity of our approach, and show its efficacy through simulations and experiments where groups of robots executing Dubins curves travel to their assigned goal locations without collisions.**

I. Introduction

The integration of robotics into everyday life is becoming increasingly more prevalent, from the use of unmanned aerial vehicles (UAVs) for package delivery [1], to the use of UAVs for the damage and wear inspection of penstocks [2]. The efficiency of these systems is further improved through the use of multi-robot teams. The collaboration of multi-robot teams to complete a higher objective often times requires individual robots to be assigned to tasks, and for these robots to then maneuver within their environment. In particular, we are interested in the problem of assigning individual robots to goal destinations, and generating the navigation plans that reach these destinations while ensuring there are no collisions.

Solutions to this problem are useful for applications where teams of robots must be assigned, and safely guided, to tasks at various locations. An example of this includes the assignment of robots to destinations for search and rescue missions [3] [4]. Similarly, multi-robot motion planning is important for surveillance when ensuring the safety of a given region [5]. Other examples include the allocation of mobile robots to fetch products in automated fulfillment centers [6], and the coordination of in-motion formation changes [7].

This paper presents a solution to the multi-robot motion planning problem for teams of Dubins cars. This problem is, given a team of Dubins cars and a set of unlabeled goal locations, assign each Dubins car to a goal location and generate collision-free trajectories to these goal locations. We are currently only concerned about getting each Dubins car to a goal location, not with having it arrive with any specific orientation; in other words, we consider oriented starts and orientation-free goals. The trajectories must satisfy the kinematic constraints on the motion of each Dubins car, while also ensuring that there are no collisions. These kinematic constraints include the orientation and minimum turning radius of the Dubins cars.

In this paper, we are concerned about developing a solution to this problem that is computationally tractable. For this reason, we present a solution that concurrently solves the assignment and planning of trajectories for teams of Dubins cars (DC-CAPT). Coupling the assignment and trajectory planning subproblems allows for a more combinatorially efficient solution. Additionally, we will show that collision avoidance in this coupled solution can be ensured through the use of a geometric constraint on the initial set-up of the Dubins cars and goals.

There are a number of challenges to solving the problem presented in this paper. The kinematic constraints on the Dubins cars' motion often prevent it from moving directly toward its goal. Additionally, Dubins curves (as will be presented below) are piecewise nonlinear curves which make deriving analytical results challenging. Lastly, the length and path of Dubins curves are dependant on the Dubins car's orientation and Euclidean distance to the goal. This makes the trajectories from a robot to each goal unique. Thus when considering more than one Dubins car, acquiring an assignment and planning collision-free trajectories becomes nontrivial.

II. Related Work

There are a number of methods that may be employed to solve the problem presented in this paper. Early work by Kloder et al. presents a solution to this problem that uses the roots of complex polynomials to specify configurations (formations) of robots [8]. This work is focused on changing formations of holonomic robot teams using these permutation-invariant polynomials, and thus is not capable of handling the kinematic constraints of Dubins cars.

Alonso-Mora et al. in [9] and [10] propose a method for solving the multi-robot motion planning problem using reciprocal collision avoidance for teams of nonholonomic robots. Similarly, Kirkpatrick et al. study the shortest collision-avoidance motion for two discs in [11]. In both solutions robots are commanded to deviate from their optimal paths. Additionally, completeness is not guaranteed when using reciprocal collision avoidance since deadlocks may arise.

We gratefully acknowledge the support of ARL grant DCIST CRA W911NF-17-2-0181, NSF grants CNS-1521617, ARO grant W911NF-13-1-0350.

Michael Whitzer, Daigo Shishika, Dinesh Thakur, and Vijay Kumar are with the GRASP Lab at the University of Pennsylvania, USA,{mwhitzer|shishika|tdinesh|kumar} @seas.upenn.edu. Amanda Prorok is with the Department of Computer Science and Technology, University of Cambridge, UK, asp45@cam.ac.uk.

An assignment solution for groups of unlabeled discs in an obstacle filled environment can be found in work done by Turpin et al. [12]. This work finds an assignment that minimizes the maximum cost over all robot trajectories. A similar work was done by Solovey et al. [13] which seeks to minimize the sum of the path lengths of each robot. The former avoids collisions through prioritization, and the latter by having robots stray from their desired paths. In each solution, the trajectory planning requires additional computational complexity in order to ensure collision avoidance.

The theoretical foundation of our work was first proposed by Turpin et al., in a method called the Concurrent Assignment and Planning of Trajectories (CAPT) algorithm [14]. The strength of this method is that it provides a complete assignment to goal positions with collision-free, optimal time-parameterized trajectories for all robots in the system. The CAPT algorithm, however, is focused solely on the use of holonomic robots that execute straight-line paths, and thus is not directly capable of providing the same guarantees for robots with motion constraints.

Contributions: The contributions of this work are three-fold: *(i)* we prove that a geometric condition exists for teams of Dubins cars that guarantees collision avoidance for our coupled assignment and trajectory planning solution; *(ii)* we provide an analytical expression for this geometric condition; and *(iii)* we provide a numerical method to find an even tighter geometric condition than our analytical expression. To the best of our knowledge, this is the first work to concurrently solve assignment and trajectory generation for teams of Dubins cars while guaranteeing collision avoidance, and therefore provides a baseline for further development.

III. PRELIMINARIES

This section presents the notational foundation for the solution presented in this paper for the assignment of Dubins cars. It is important to note that while the kinematic model presented here is for car-like nonholonomic robots (Dubins car), DC-CAPT is applicable to the class of nonholonomic robots whose trajectories are constrained by a minimum curvature.

The notation used in this paper is analogous to that found in [14]. A 2-dimensional Euclidean space is considered where there are N homogeneous Dubins cars of radius R, and M goal locations. The state of the ith Dubins car is given as $\mathbf{x}_i(t) = [\mathbf{p}_i(t), \theta_i(t)]^\top$ for $i = \{1, \dots N\}$ where at time t, $\mathbf{p}_i(t) = [\,x_i(t), y_i(t)]^\top$ defines the location of the Dubins car and $\theta_i(t) \in [0, 2\pi)$ is its orientation. When a distinction is needed between the location of a robot on a linear trajectory and the location of a robot on a Dubins curve, $\mathbf{p}_i^L(t)$ and $\mathbf{p}_i^D(t)$ will be used, respectively.

The jth goal location is denoted as $\mathbf{g}_j = [x_j, y_j]^\top$ for $j = \{1 \dots M\}$. The state vector, $\mathbf{X}(t) \in \mathbb{SE}(2)^N$, of the system of Dubins cars is $\mathbf{X}(t) = [\,\mathbf{x}_1(t)^\top, \mathbf{x}_2(t)^\top, \dots \mathbf{x}_N(t)^\top]^\top$. The *assignment matrix*, $\boldsymbol{\phi} \in \mathbb{R}^{N \times M}$ that assigns robots to goals is

$$\phi_{ij} = \begin{cases} 1 & \text{if robot } i \text{ is assigned to goal } j \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

In order to ensure that all Dubins cars (or goals) are assigned, the following conditions should hold

$$\begin{aligned} \boldsymbol{\phi}^\top \boldsymbol{\phi} &= \mathbf{I}_M \quad \text{if } N \geq M, \\ \boldsymbol{\phi}\boldsymbol{\phi}^\top &= \mathbf{I}_N \quad \text{if } N \leq M. \end{aligned} \tag{2}$$

In the case where $N = M$, $\mathbf{I}_N = \mathbf{I}_M$ is the identity matrix. DC-CAPT finds trajectories $\boldsymbol{\gamma}(t) : [t_0, t_f] \to \mathbb{SE}(2)^N$ where t_0 and t_f are the initial and final times, respectively. The initial conditions at t_0 are defined as

$$\mathbf{X}(t_0) = [\,\mathbf{x}_1(t_0)^\top, \mathbf{x}_2(t_0)^\top, \dots \mathbf{x}_N(t_0)^\top]^\top, \tag{3}$$

where the initial orientation of each robot will be denoted as $\alpha_i = \theta_i(t_0)$. Similarly, the goal conditions at t_f are

$$\begin{aligned} \Phi^\top \boldsymbol{\gamma}(t_f) &= \mathbf{X}(t_f) \quad \text{if } N \geq M, \\ \boldsymbol{\gamma}(t_f) &= \Phi \mathbf{X}(t_f) \quad \text{if } N < M, \end{aligned} \tag{4}$$

where $\Phi = \boldsymbol{\phi} \otimes \mathbf{I}_3$ is the expanded assignment matrix, with $\otimes$ representing the Kronecker product. Note that $\mathbf{p}_i(t_f) = \mathbf{g}_j$ for the ith robot assigned to jth goal.

The kinematic model, assuming no slip, for a Dubins car is captured by the bicycle model:

$$\begin{bmatrix} \dot{x}_i(t) \\ \dot{y}_i(t) \\ \dot{\theta}_i(t) \end{bmatrix} = v_i \begin{bmatrix} \cos(\theta_i(t)) \\ \sin(\theta_i(t)) \\ \frac{1}{b_i} \tan \psi_i \end{bmatrix} \tag{5}$$

where v_i, b_i and ψ_i are the linear velocity, wheelbase and steering angle of the ith Dubins car, respectively. In this paper, we assume that the Dubins cars travel at constant speed, however, individual Dubins car speeds may differ.

The necessary and sufficient condition to assure collision avoidance is

$$\delta_{ij}(t) = \| \mathbf{p}_j(t) - \mathbf{p}_i(t) \| > 2R, \forall i, j \in \{1, \dots, N\}, \tag{6}$$

for $t \in [t_0, t_f]$. Lastly, note that we normalize the time such that $t_0 = 0$ and $t_f = 1$, thus $t \in [0, 1]$.

A. Dubins Curves

We use Dubins curves because they offer optimal trajectories between oriented starts and goals, and provide approximations for more sophisticated vehicles. They are bounded-curvature shortest-path curves [15] [16] that are composed of two motion primitives; turning (C) and going straight (S). These motion primitives make up the sets of curves written as CSC or CCC, however, in this work we will only be concerned with the subset of curves written as CS.[1] This is because we are only concerned about getting each Dubins car to a goal location, not with having it arrive with any specific orientation; oriented starts with orientation-free goals.

The set of CS curves can be classified into two curves represented by $\mathcal{D} = \{RS, LS\}$, where RS represents right-straight curves, and LS represents left-straight curves. The turning motion primitive (C) is constrained by a minimum turning radius, ρ_i. In this paper we assume that all Dubins cars have equal turning radii, therefore, $\rho_i = \rho$. Additionally,

[1]The CCC Dubins curves, and its subset of CC curves, will not be considered in this work.

the nondimensionalized turning radius will be denoted by $\hat{\rho} = \rho/R$.

The length of each Dubins curve in $\mathcal{D}$ between $\mathbf{p}_i(t_0)$ and $\mathbf{g}_j$ is the sum of the lengths of each motion primitive. The length of the turning motion primitive (C) is $c_{ij}(l_{ij}, \alpha_i)$, and the length of the straight motion primitive (S) is $s_{ij}(l_{ij}, \alpha_i)$, where $l_{ij} = ||\mathbf{p}_i(t_0) - \mathbf{g}_j||$, is the Euclidean distance between the ith Dubins car's starting location and the jth goal. Therefore, the total length of a CS Dubins curve is $L_{ij}(l_{ij}, \alpha_i) = c_{ij}(l_{ij}, \alpha_i) + s_{ij}(l_{ij}, \alpha_i)$.

IV. Problem Definition

Our problem considers the assignment and trajectory generation of teams of Dubins cars to unlabeled goal locations. We seek a solution that is computationally tractable and that satisfies the kinematic constraints of Dubins cars while ensuring there are no collisions. To solve this problem, DC-CAPT concurrently solves the assignment and trajectory generation subproblems through

$$\begin{aligned} \underset{\boldsymbol{\phi}}{\text{maximize}} \quad & \textstyle\sum_{i=0}^{N} \sum_{j=0}^{M} \phi_{ij} J_{ij} \\ \text{subject to} \quad & (1)\text{: Valid assignments} \\ & (2)\text{: Full resource utilization} \\ & (3)\text{: Initial conditions} \\ & (4)\text{: Terminal conditions} \\ & (5)\text{: Robot capabilities} \\ & (6)\text{: Collision avoidance,} \end{aligned} \tag{7}$$

for some cost J_{ij} under the following assumptions:

- **(A1)** All robots are homogeneous and interchangeable.
- **(A2)** Robots are car-like and represented as discs of radius R and have turning radius ρ.
- **(A3)** The environment is obstacle-free.
- **(A4)** All robots have perfect knowledge of their state and there are no errors in actuation.
- **(A5)** $\mathbf{x}_i(t_0)$ and $\mathbf{g}_j$ are placed with separation distance Δ apart:

$$\begin{gathered} \| \mathbf{p}_i(t_0) - \mathbf{p}_j(t_0) \| > \Delta, \forall i \neq j \in \{1, \ldots N\} \text{ , and} \\ \| \mathbf{g}_i - \mathbf{g}_j \| > \Delta, \forall i \neq j \in \{1, \ldots, M\} \text{ , and} \\ \| \mathbf{p}_i(t_0) - \mathbf{g}_j \| > \Delta, \forall i \in \{1, \ldots, N\}, j \in \{1, \ldots, M\}. \end{gathered} \tag{8}$$

In this work, we seek to find a solution to (7) where the trajectories are CS Dubins curves that connect oriented starts with orientation-free goals. As noted earlier, our work builds upon [14] where collision avoidance for the optimal assignment (discussed later) was made possible by finding a safe separation distance Δ. As in [14], the assignment specifies the optimal trajectories, $\gamma^*(t)$, because they are deterministic given start and goal states. The focus of this paper will be to find a safe separation distance Δ such that collision avoidance is guaranteed. The theory used to find this Δ will be simplified by specifying that all Dubins cars must simultaneously arrive at their assigned goals. This is a valid specification given that synchronization is imperative for multi-robot applications such as formation control, as was mentioned earlier.

Definition 1 (Collision-Free Assignment)**.** *A collision-free assignment is an assignment of N Dubins cars with starting states $\mathbf{x}_i(t_0)$, radius R, and turning radius ρ, to M goal locations $\mathbf{g}_j$, where $i \in \{1 \ldots N\}$ and $j \in \{1 \ldots M\}$, such that all paths satisfy* (6) *when considering the simultaneous arrival of Dubins cars at their assigned goals.*

The solution to (7) is ensured to be collision-free by solving the challenge presented in Problem 1 below.

Problem 1. *Given a set of N Dubins cars of radius R with a turning constraint ρ, find a separation distance Δ that guarantees collision-free assignment for the simultaneous arrival of Dubins cars to their assigned goals while satisfying* (7).

The solution to Problem 1 is challenging due to the kinematic constraints of Dubins cars. The reachability set at a given time t is no longer a single location along a line connecting the start $\mathbf{p}_i(t_0)$ and goal $\mathbf{g}_j$ as in [14]. The reachability set is now a range of possible Dubins car states $\mathbf{x}_i(t)$ that depend on α_i and l_{ij}, making it difficult to ensure collision avoidance. Additionally, these Dubins curve trajectories are piecewise nonlinear curves which make finding a closed-form solution to Problem 1 nontrivial.

Although we build upon the results of [14] to find a similar separation distance constraint (Δ), the development to find Δ given the above challenges makes this work nonincremental. Our methodology for finding bounds on Δ addresses the challenges mentioned herein and is outlined below.

V. Methodology

This section outlines the various components and development of the DC-CAPT algorithm. First, the optimal assignment is shown for which we find the minimum distance between two holonomic robots executing linear trajectories at any moment in time. An inflated robot radius is then found and used to find an analytical expression that defines a sufficient condition for Δ. This is then used as an upper bound in a method that utilizes a time-varying inflated robot radius to find an even tighter bound on Δ.

A. Optimal Assignment

The *optimal assignment* chosen for DC-CAPT is the one that minimizes the sum of the squared Euclidean distances between the oriented starts and orientation-free goal locations. This is the same assignment used in CAPT [14], and thus we are able to build upon the results of [14] to develop our solution.

The optimal assignment is found by first constructing a distance matrix, $\mathbf{D}$, that stores the ith Dubins car's squared Euclidean distance to the jth goal; $D_{ij} = l_{ij}^2$. This is then used in

$$\boldsymbol{\phi}^* = \underset{\boldsymbol{\phi}}{\text{argmin}} \sum_{i=0}^{N} \sum_{j=0}^{M} \phi_{ij} D_{ij} \tag{9}$$

to find the optimal assignment, $\boldsymbol{\phi}^*$. This optimization problem can be solved using the Hungarian algorithm in $\mathcal{O}(N^3)$ time [17].

B. Minimum Holonomic Robot Proximity Throughout Linear Trajectory Execution

It was shown in Lemma 3.2 of [14] that optimal assignments (robot i assigned to goal i, and robot j assigned to goal j) satisfy $\mathbf{w}_{ij}^{\top}\mathbf{u}_{ij} \geq 0$ where $\mathbf{w}_{ij} = \mathbf{p}_j^L(t_f) - \mathbf{p}_i^L(t_f)$ and $\mathbf{u}_{ij} = \mathbf{p}_j^L(t_0) - \mathbf{p}_i^L(t_0)$. This result was used to find the closest distance between two holonomic robots executing linear trajectories; this occurs when $\mathbf{w}_{ij}^{\top}\mathbf{u}_{ij} = 0$. The squared distance between these robots for $t \in [0, 1]$ was shown in [14] to be

$$||\mathbf{p}_j^L(t) - \mathbf{p}_i^L(t)||^2 = a - 2t(a-b) + t^2(a - 2b + c), \quad (10)$$

where $a = \mathbf{u}_{ij}^{\top}\mathbf{u}_{ij}$, $b = \mathbf{w}_{ij}^{\top}\mathbf{u}_{ij}$, and $c = \mathbf{w}_{ij}^{\top}\mathbf{w}_{ij}$. The desired collision avoidance constraint from (6) is $||\mathbf{p}_j^L(t) - \mathbf{p}_i^L(t)|| > 2R$, and is shown in [14] to be achieved for $t \in [0, 1]$ when $a > 8R^2$, $b = 0$ and $c > 8R^2$ (worst case), and thus $\Delta > 2\sqrt{2}R$. This means that two holonomic robots executing linear trajectories between starts and goals will not get closer than $2R$ from one another for $t \in [0, 1]$ if $\Delta > 2\sqrt{2}R$.

Lemma 1. *Robots executing linear trajectories for an optimal assignment will satisfy*

$$||\mathbf{p}_j^L(t) - \mathbf{p}_i^L(t)|| > \Delta\sqrt{1 - 2t + 2t^2} \ \forall \ t \in [0, 1]. \quad (11)$$

Proof. Using the worst case parameters $a > 8R^2$, $b = 0$ and $c > 8R^2$ in (10), we get $\Delta^2(1 - 2t + 2t^2)$. It must be shown for any a, b and c, and for all $t \in [0, 1]$, that the following holds;

$$a - 2t(a-b) + t^2(a - 2b + c) > \Delta^2(1 - 2t + 2t^2) \ \forall \ t \in [0, 1]. \quad (12)$$

Rearranging the above inequality gives

$$(a - \Delta^2) - 2t(a - b - \Delta^2) + t^2(a - 2b + c - 2\Delta^2) > 0 \ \forall \ t \in [0, 1],$$

which is true if the discriminant is negative; $b^2 - (a - \Delta^2)(c - \Delta^2) < 0$. To determine if this is valid, it is important to ensure that $t \in [0, 1]$, leading to $0 \leq (a - b - \Delta^2)/(a - 2b + c - 2\Delta^2) \leq 1$, where it can be seen that $a - 2b + c \geq 2\Delta^2$. This leads to the constraints $a - \Delta^2 \geq b$ and $c - \Delta^2 \geq b$, which when applied to the discriminant, shows that it is negative for any a, b and c because $b^2 \leq (a - \Delta^2)(c - \Delta^2)$. Therefore, (12) holds and thus taking the square root of both sides gives

$$||\mathbf{p}_j^L(t) - \mathbf{p}_i^L(t)|| > \Delta\sqrt{1 - 2t + 2t^2} \ \forall \ t \in [0, 1].$$

□

C. Inflated Robot Radius

If we can approximate the motion of a Dubins car as a holonomic robot executing a linear trajectory, we can use the results above to ensure collision avoidance for teams of Dubins cars. This approximation is done through the use of an inflated robot radius, $\tilde{R}$ (defined below). This inflated robot radius is then used to find an analytical expression for Δ that guarantees the collision-free execution of the optimal assignment.

Definition 2 (Inflated Robot Radius)**.** *Consider two robots where one executes a Dubins curve with position denoted by $\mathbf{p}^D(t)$, and the other executes a linear trajectory with position denoted by $\mathbf{p}^L(t)$, for time $t \in [0, 1]$. These robots start and finish their executions at the same locations and same time.*[2] *The inflated robot radius is then a bound $\tilde{R}$ such that $R + \|\mathbf{p}^L(t) - \mathbf{p}^D(t)\| \leq \tilde{R}$ for all start and goal states, and all t.*

Any robot executing a Dubins car trajectory between an oriented start and an orientation-free goal location stays within a disc of radius $\tilde{R}$ that moves along a linear trajectory between the same start and goal. This means that we can approximate a Dubins car as a holonomic robot of radius $\tilde{R}$ that executes a linear trajectory. The results found in [14] for CAPT can then be used to establish the following theorem.

Theorem 1. *If $\Delta > 2\sqrt{2}\tilde{R}$, with $\tilde{R} = \rho\sqrt{\pi^2 + 4} + R$, then any optimal assignment is safely executed by Dubins cars of true radius R and turning radius ρ.*

Proof. In order to find $\tilde{R}$, we will first find the maximum deviation of a Dubins curve from a straight line trajectory between a start and goal.

Consider the starting state $(0, 0, \alpha)$ and goal location $(l, 0)$ with the xy-coordinate system oriented such that the x-axis is in line with the goal. The deviations of a Dubins curve from a straight-line trajectory between this start and goal will be described by $\tilde{\mathbf{x}}(t) = \mathbf{p}^L(t) - \mathbf{p}^D(t)$, where $\tilde{\mathbf{x}}(t) = [\,\tilde{x}_x(t), \tilde{x}_y(t)]^{\top}$. Here $\mathbf{p}^D(t)$ represents the position of a Dubins car on the turning portion (C) of a CS Dubins curve.[3] This is the only portion of the CS Dubins curve that matters in the analysis used to find $\tilde{R}$. This is because $\tilde{R}$ represents the worst case deviation of a Dubins curve from a straight-line trajectory. During C a Dubins car can travel opposite the direction of a linear robot before heading in the direction of the goal, therefore maximizing the deviation from a straight-line trajectory. During S the Dubins car is always heading in the direction of the goal.

Maximizing the x and y components of $\tilde{\mathbf{x}}(t)$ is captured by

$$R_x = \underset{t}{\text{maximize}}\, |\tilde{x}_x(t)| \text{ and } R_y = \underset{t}{\text{maximize}}\, |\tilde{x}_y(t)|. \quad (13)$$

To find R_x, note that the x-coordinates of the linear robot and the Dubins car between the start and goal are

$$x^L(t) = tl \text{ and } x^D(t) = \rho\left(-\sin\left(\alpha - \frac{tL}{\rho}\right) + \sin\alpha\right),$$

where $L = L(l, \alpha)$. Plugging $x^L(t)$ and $x^D(t)$ into $\tilde{\mathbf{x}}(t)$ gives

$$\tilde{x}_x(t) = tl - \rho\left(-\sin\left(\alpha - \frac{tL}{\rho}\right) + \sin\alpha\right). \quad (14)$$

Taking the derivative of $\tilde{x}_x(t)$ with respect to t, setting this equal to zero, solving for t, and then plugging this back into

[2]Subscripts i and j are not used on variables when doing analysis on a specific start and goal pair.

[3]CS Dubins curves are symmetric about the axis that is collinear with the line connecting a start and goal, so we will restrict our analysis in this paper to RS Dubins curves. The same analysis holds for LS Dubins curves.

(14) gives

$$\tilde{x}_x(t) = \rho\left(\frac{l}{L}\left(\alpha - \cos^{-1}\left(\frac{l}{L}\right)\right) + \sqrt{1 - \left(\frac{l}{L}\right)^2} - \sin\alpha\right),$$

which has a maximum value of $R_x = \pi\rho$ when $\alpha = \pi$ and as $\frac{l}{L} \to 1$.

Finding R_y follows the same procedure where from $\tilde{\mathbf{x}}(t)$ we have

$$\tilde{x}_y(t) = -\rho\left(\cos\left(\alpha - \frac{tL}{\rho}\right) - \cos\alpha\right). \tag{15}$$

Taking the derivative of $\tilde{x}_y(t)$ with respect to t, setting this equal to zero, solving for t, and then plugging this back into (15) gives a maximum value of $R_y = 2\rho$ when $\alpha = \pi$.

Given R_x and R_y, we can find $\tilde{R}$ through $\tilde{R} = \sqrt{R_x^2 + R_y^2} + R$, and thus

$$\tilde{R} = \rho\sqrt{\pi^2 + 4} + R. \tag{16}$$

This $\tilde{R}$ can then be used in $\Delta > 2\sqrt{2}\tilde{R}$ to guarantee the safe execution of any optimal assignment given the results from [14]. □

The above provides a closed-form expression for $\tilde{R}$ and a collision avoidance guarantee for the concurrent assignment and trajectory planning of teams of Dubins cars. It should be noted, however, that $2\sqrt{2}\tilde{R}$ provides a conservative bound on Δ. This is because the deviation of a Dubins car from a straight-line trajectory is a time varying value. The result in (16) is constant for all $t \in [0, 1]$, therefore for some times, it is going to be conservative. Hence we try to reduce this over conservativeness by using a time-varying inflated radius.

D. Time-Varying Inflated Robot Radius

A time-varying inflated robot radius, $\tilde{R}(t)$, represents the maximum deviation of a Dubins car from a straight-line trajectory at a particular time $t \in [0, 1]$. It represents the worst case scenario at time t, and must satisfy the following definition.

Definition 3 (Time-Varying Inflated Robot Radius)**.** *Consider a robot executing a Dubins curve with position denoted by $\mathbf{p}^D(t)$, and a robot executing a linear trajectory with position denoted by $\mathbf{p}^L(t)$, for time $t \in [0, 1]$. These robots start and finish their executions at the same locations and same time. The time-varying inflated robot radius is then a bound $\tilde{R}(t)$ such that $R + \|\mathbf{p}^L(t) - \mathbf{p}^D(t)\| \leq \tilde{R}(t)$ for all start and goal states at a particular time t.*

Any robot executing a Dubins car trajectory between an oriented start and an orientation-free goal location stays inside a disc of radius $\tilde{R}(t)$ that is moving along a linear trajectory between the same start and goal. This means that at time t, the radius of a holonomic robot executing a linear trajectory would have to be at least $\tilde{R}(t)$ to contain any location a Dubins car could be in at the same time t.

In order to find $\tilde{R}(t)$ at time t, we will assume for now that Δ is known. Additionally, given the piecewise nonlinear nature of CS Dubins curves, $\tilde{R}(t)$ must be found along the turning (C) and straight (S) portions, depending on the value of t. It was numerically found that $\alpha = \pi$ is the worst case configuration for all values of t, results in the maximum amount of turning, and thus this is the orientation used to find $\tilde{R}(t)$.

The optimization problem used to find $\tilde{R}(t)$ is thus

$$\begin{aligned} \tilde{R}(t) \quad &= \max_l \bar{R}(t, l) \\ \text{Subject to} \quad &\Delta \leq l \end{aligned} \tag{17}$$

for

$$\bar{R}(t, l) = \begin{cases} \bar{R}_C(t, l) + R & \text{if } t \in [0, t_C(l)] \\ \bar{R}_S(t, l) + R & \text{otherwise,} \end{cases}$$

where $\bar{R}_C(t, l)$ represents the maximum deviation of a Dubins car from a straight-line trajectory along the turning (C) portion and is defined as

$$\bar{R}_C(t, l) = \sqrt{l^2t^2 + 2l\rho t \sin\left(\frac{tL}{\rho}\right) - 2\rho^2(\cos\left(\frac{tL}{\rho}\right) - 1)},$$

and $\bar{R}_S(t, l)$ represents this maximum deviation along the straight (S) portion and is defined as

$$\begin{aligned} \bar{R}_S(t, l) = \quad & \tfrac{1-t}{1-t_C(l)}\sqrt{(-\rho\sin(\tfrac{c}{\rho}) - l + \tfrac{l^2}{L})^2 + \ldots} \\ & \overline{\ldots(\rho - \rho\cos(\tfrac{c}{\rho}))^2}, \end{aligned}$$

with $c = c(l, \pi) = \rho(\pi + 2\tan(\rho/l))$ and $L = L(l, \pi) = l + c(l, \pi)$ (derived from the geometry of a CS Dubins curve when $\alpha = \pi$), and $t_C(l) = c(l, \pi)/L(l, \pi)$ is the maximum possible time on the turning portion (C).

E. Compute Separation Distance, Δ

Now that we know how to find $\tilde{R}(t)$ at a particular time t, we are able to find a tighter Δ than that presented in Theorem 1. This tighter Δ will be found using the bisection method, which will compute Δ within a specified error bound. The minimum bound used in the bisection method is $\Delta_{min} = 4\rho + 2R$, which is the physical limit on the proximity between two CS Dubins curves. This physical limit ensures that two Dubins cars cannot turn into one another. The maximum bound used is $\Delta_{max} = 2\sqrt{2}\tilde{R}$, which is the analytical result found in Theorem 1. The method will search over values of Δ between Δ_{min} and Δ_{max}, find $\tilde{R}(t) \forall t \in [0, 1]$, and compute the error $e(t) = ||\mathbf{p}_j^L(t) - \mathbf{p}_i^L(t)|| - 2\tilde{R}(t) \forall t \in [0, 1]$. The process stops when $0 < e(t) < \epsilon$, where $\epsilon > 0$ is a desired error value.

Theorem 2. *Given an error value $\epsilon > 0$, if* (8) *is satisfied, where Δ is found using the bisection method, then any optimal assignment is guaranteed to be collision-free.*

Proof. From Lemma 1, the minimum distance between holonomic robots executing linear trajectories is known to satisfy (11). We also know from Section V-D how to find $\tilde{R}(t)$ for any time $t \in [0, 1]$. Given a Dubins car of radius R, $\tilde{R}(t)$ describes the radius of a holonomic robot executing a linear trajectory that contains any possible location of the Dubins car at time t. This can be used with (6) to give the

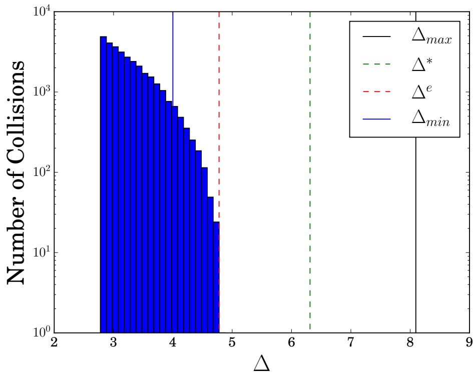

Fig. 1: The number of Dubins car collisions as a function of Δ. Each Δ was run for 10,000 trials with $\rho = 0.5$ and $R = 1$.

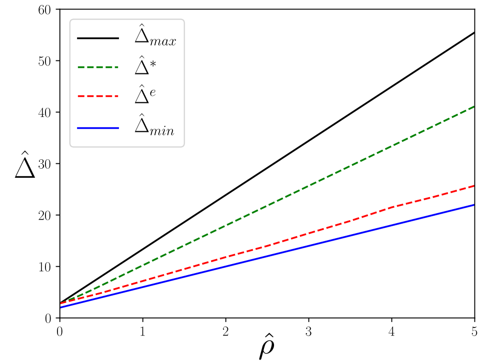

Fig. 2: Nondimensional $\hat{\Delta}$ as a function of the nondimensional turning radius ($\hat{\rho}$). $\hat{\Delta}^*$ is the bound generated by DC-CAPT, while $\hat{\Delta}^e$ is the empirically acquired $\hat{\Delta}$.

collision avoidance constraint $||\mathbf{p}_j^L(t) - \mathbf{p}_i^L(t)|| > 2\tilde{R}(t)$ for $t \in [0, 1]$. Re-arranging this leads to $||\mathbf{p}_j^L(t) - \mathbf{p}_i^L(t)|| - 2\tilde{R}(t) > 0$ for $t \in [0, 1]$, which is the constraint used in the bisection method to find Δ. The Δ found using the bisection method must satisfy this constraint to be selected, and therefore guarantees collision avoidance for the optimal assignment of teams of Dubins cars. $\square$

VI. Results

This section demonstrates the validity of DC-CAPT through various simulations and experiments. We show the relationship between Dubins car collisions and separation distance Δ, as well as the effects of turning radius ρ on Δ. Simulations were also run for teams of Dubins cars, and we demonstrate another application space of DC-CAPT through experiments with a group of quadrotors.

A. Numerical Results

The relationship between Dubins car collisions and Δ is seen in Fig. 1 for an example when $\rho = 0.5$ and $R = 1$. This data was generated by taking 10,000 random samples of starts and goals for Δ from $2\sqrt{2}R$ to $2\sqrt{2}\tilde{R}$ in increments of 0.1. These randomly sampled starts and goals were for two robots such that $b = \mathbf{w}_{ij}^\top \mathbf{u}_{ij} = 0$ (see Section V-B). It can be seen that as Δ increases, the number of collisions decreases. The upper and lower bounds on Δ are shown in the figure, along with an empirically derived separation distance labeled Δ^e, and a separation distance generated using our method labeled Δ^*. Even though $\Delta^e < \Delta^*$, there are no guarantees that collisions do not occur for $\Delta^e < \Delta < \Delta^*$. The Δ^* generated using our method is guaranteed to generate collision-free optimal assignments.

The effects of the turning radius ρ on Δ is shown in Fig. 2. An approximately linear relationship is shown between $\hat{\rho}$ and $\hat{\Delta}$ (where $\hat{\Delta} = \Delta/R$). It also shows that the difference between $\hat{\Delta}^e$ and $\hat{\Delta}^*$ decreases with decreasing $\hat{\rho}$ (conversely, it increases with increasing $\hat{\rho}$). The linear trend of $\hat{\Delta}^e$ shows that we cannot do better than a linearly increasing $\hat{\Delta}$. Additionally, although $\hat{\Delta}^e$ and $\hat{\Delta}^*$ are diverging, the conservativeness (i.e., the ratio $\hat{\Delta}^*/\hat{\Delta}^e$) remains approximately constant with increasing $\hat{\rho}$.

(a) Simulation (b) Experiment

Fig. 3: (a.) Example assignment and trajectories for five Dubins cars used in simulation and (b.) a demonstration of our experiment using three quadrotors.

B. Illustrative Examples

Simulations were run for a team of five Dubins cars using the MORSE simulator. This demonstrated the effectiveness of DC-CAPT to navigate Dubins cars to goal locations while ensuring collision avoidance. An example of an assignment and trajectories for five Dubins cars can be seen in Fig. 3a.

We demonstrate another application space for DC-CAPT through our real-robot experiments. We implemented our method for a team of three quadrotors using visual inertial odomotery (VIO) for position tracking and navigation (see Fig. 3b). In order to ensure constant speed object detection, Dubins curves can be used between starts and goals. The quadrotors in our experiment were able to safely navigate to their assigned goal locations.

VII. Conclusion

In this paper we have shown a method for the concurrent assignment and planning of trajectories for teams of Dubins cars (DC-CAPT) that guarantees collision avoidance. This work builds upon the results of [14] to find a geometric constraint (safe separation distance), Δ, that provides this collision-free assignment guarantee. Our method utilizes Dubins curves as optimal trajectories and an inflated time-varying radius to ensure that all Dubins cars are safe during the execution of their trajectories. Furthermore, our solution is computationally tractable, having complexity $\mathcal{O}(N^3)$. Lastly, this methodology is applicable to the case when we also have oriented goals, which is a topic of future work.

REFERENCES

[1] Forbes. Meet amazon prime air, a delivery-by-aerial-drone project, Dec. 2013.

[2] Tolga Özaslan, Shaojie Shen, Yash Mulgaonkar, Nathan Michael, and Vijay Kumar. *Inspection of Penstocks and Featureless Tunnel-like Environments Using Micro UAVs*, pages 123–136. Springer International Publishing, Cham, 2015.

[3] Yanli Yang, Marios M Polycarpou, and Ali A Minai. Multi-UAV Cooperative Search Using an Opportunistic Learning Method. *Journal of Dynamic Systems, Measurement, and Control*, 129(5):716–728, 2007.

[4] George Kantor, Sanjiv Singh, Ronald Peterson, Daniela Rus, Aveek Das, Vijay Kumar, Guilherme Pereira, and John Spletzer. Distributed search and rescue with robot and sensor teams. In *Field and Service Robotics*, pages 529–538. Springer, 2006.

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

[6] J Enright and P R Wurman. Optimization and Coordinated Autonomy in Mobile Fulfillment Systems. *Automated action planning for autonomous mobile robots: Papers from the 2011 AAAI Workshop*, 2011.

[7] Michael Whitzer, James Keller, Subhrajit Bhattacharya, Vijay Kumar, Trevor Sands, Lee Ritholtz, Adrian Pope, and Dean Dickmann. In-flight formation control for a team of fixed-wing aerial vehicles. In *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 372–380. IEEE, 2016.

[8] S. Kloder and S. Hutchinson. Path planning for permutation-invariant multirobot formations. *IEEE Transactions on Robotics*, 22(4):650–665, Aug 2006.

[9] Javier Alonso-Mora, Andreas Breitenmoser, Martin Rufli, Roland Siegwart, and Paul A. Beardsley. Image and animation display with multiple mobile robots. *I. J. Robotics Res.*, 31:753–773, 2012.

[10] J. Alonso-Mora, A. Breitenmoser, P. Beardsley, and R. Siegwart. Reciprocal collision avoidance for multiple car-like robots. In *2012 IEEE International Conference on Robotics and Automation*, pages 360–366, May 2012.

[11] David G. Kirkpatrick and Paul Liu. Characterizing minimum-length coordinated motions for two discs. *CoRR*, abs/1607.04005, 2016.

[12] Matthew Turpin, Kartik Mohta, Nathan Michael, and Vijay Kumar. Goal assignment and trajectory planning for large teams of interchangeable robots. *Auton. Robots*, 37(4):401–415, 2014.

[13] Kiril Solovey, Jingjin Yu, Or Zamir, and Dan Halperin. Motion planning for unlabeled discs with optimality guarantees. In *Robotics: Science and Systems XI, Sapienza University of Rome, Rome, Italy, July 13-17, 2015*, 2015.

[14] Matthew Turpin, Nathan Michael, and Vijay Kumar. Capt: Concurrent assignment and planning of trajectories for multiple robots. *The International Journal of Robotics Research*, 33(1):98–112, 2014.

[15] Steven M. LaValle. *Planning Algorithms*. Cambridge University Press, New York, NY, USA, 2006.

[16] 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*, 79(3):497–516, 1957.

[17] H. W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.