

---

# DESIGN OF PLANAR COLLISION-FREE TROCHOIDAL PATHS FOR A MULTI-ROBOT SWARM

---

A PREPRINT

**Adil Shiyas**

International Institute of Information Technology Bangalore  
Bangalore, India  
adil.shiyas@iiitb.ac.in

**Sachit Rao**

International Institute of Information Technology Bangalore  
Bangalore, India  
sachit@iiitb.ac.in

November 21, 2023

## ABSTRACT

In the literature, a distributed consensus protocol by which a connected swarm of agents can generate artistic patterns in 2-dimensional space is proposed. Motivated by this protocol, in this paper, we design the parameters of this protocol for a 3-agent swarm of non-holonomic robots of finite size that results in the generation of periodic trochoidal trajectories that satisfy a set of geometric and speed constraints; this design also includes selecting the initial positions of the robots. This problem finds applications in persistent surveillance and coverage, guarding a region of interest, and target detection. While the trajectories may be self-intersecting, imposing geometric constraints *i.* eliminates collisions between robots; *ii.* ensures minimum and maximum separation distance between any robot and a fixed point, thus ensuring the robots are in communication range. Imposing speed constraints ensure that tracking these trajectories becomes feasible. It is also shown that robots can be injected to these paths at specific locations, in order to increase the *refresh rate*, without violating any of the geometric constraints. The designs are implemented in an indoor mobile robot platform.

**Keywords** Swarm Robotics; Trochoidal Patterns; Persistent Coverage; Surveillance

## 1 Introduction

The distributed consensus protocol (CP) presented in [1] enables a connected swarm of agents, modelled as single integrators, to trace repeated geometric paths in both 2-dimensional (2-D) and 3-D spaces. The parameters of the protocol, the connection topology of the swarm, and the initial positions of the agents define the characteristics of the generated paths. The applications of agents, or robots, tracing such paths have been well identified in the literature, for example, persistent coverage and surveillance of a region, guarding an asset, and target detection; the cited references provide an exhaustive list. In such applications, the issues of collisions between robots, communication range, feasibility of path tracking, and time taken to trace the path, have to be considered explicitly. Further, in applications involving guarding a specific region or an asset, the path should be defined by making the asset the centre of rotation (CoR).

In this paper, the protocol is designed for a connected swarm of 3 unicycle-robots of finite size moving in a 2-D Cartesian space; the use of 3 robots leads to the generation of trochoidal paths, where each robot traces a unique trochoid. For a given communication topology, design of the trochoidal paths involves the selection of 3 scalars for the CP, a positive integer,  $k$ , with magnitude  $k \geq 2$ , and the initial coordinates of the 3 robots in  $X - Y$  space, thus leading to a total of 10 design variables. The 3 scalars in the CP and  $k$  determine the nature of the trochoid - hypotrochoid or

epitrochoid - and the number of cusps, respectively; it is remarked that for a chosen set of these variables, all robots trace the same type of trochoid. In addition to these variables, the initial positions influence the coordinates of the common CoR, distances between robots, and the closest and farthest distances of the robots from the CoR. Thus, the constraints of no collisions or maintaining a communication range can be ensured for an appropriate choice of the design variables.

By analysing the properties of the CP and the common features of the resulting trochoids, the exercise of designing 10 variables is reduced to designing 4 variables: *i.* one scalar that determines the number of cusps ( $k$ ); *ii.* one set of positive integers that form a Pythagorean triple; and *iii.* a pair of distance metrics, denoted by  $(R_c, d_c)$ . Since the trochoids have to satisfy numerous constraints, first, these constraints are expressed in the form of linear inequalities in terms of  $(R_c, d_c)$  and second, for a choice of  $k$  and a Pythagorean triple, regions bounded by these inequalities in the  $(R_c, d_c)$  space are identified. Finally, any  $(R_c, d_c)$  pair that lies in these feasible regions is selected, which in turn, decides the values of the original design variables. By selecting a design variable as a Pythagorean triple, the trochoids become closed, [2], that is, the robots return to their respective initial positions after one period.

The use of trochoidal paths, for diverse applications, has received attention in the literature. In [2], a consensus-based control policy - and a single scalar - is proposed by which a connected network of agents defined as double integrators, all trace trochoidal trajectories with the same parameters in the 2-D space; the choice of initial conditions ensures collision avoidance. This solution is extended to the 3-D case in [3], but, collision avoidance is not explicitly considered in the analysis. In [4] and [5], the eigenvalues of a design matrix are evaluated so that agents modeled as single and double integrators, respectively, trace trochoidal paths; even in these references, collision avoidance between agents is not explicitly considered. The CP adopted in this paper guarantees eigenvalues with the desired properties; their selection is made simpler still with the use of Pythagorean triples. In [6], a feedback control law is designed using Barrier Lyapunov Functions that allow a swarm of agents, moving at constant speed, to trace closed curves, such as trochoids (also the same for all agents); the control law balances the curve-phases of the agents in the complex plane. A cyclic pursuit based control policy is proposed in [7], where the agents can trace either circular or logarithmic spiral patterns, based on the choice of a design parameter. A variant to trochoidal paths is the tracing of Lissajous curves, as described in [8, 9]; by selecting the initial locations of the agents and the parameters of the Lissajous curve (which are naturally smooth), the agents guarantee sensor coverage of a rectangular region without colliding with each other. A guidance law is proposed in [10], whose design parameter leads to a robot generating a trochoidal trajectory around a stationary target; the trochoidal trajectory is designed to maximise observability of the target based on range measurements. Bifurcation is used as the idea to generate limit cycles that are trochoidal patterns in [11]; in both [10] and [11], a single unicycle robot is considered. In this paper, the 3 agents trace different trajectories, albeit of the same type, that is, hypotrochoidal or epitrochoidal; further, these results form a design methodology, along with extensions, to the protocol proposed in [1].

From an implementation perspective, it is known that the properties of the trochoidal paths are closely tied to the initial positions. Thus, if the robots' initial positions are slightly perturbed, the resulting paths will be different and may also violate a few of the distance-based constraints. In this paper, a bound on such perturbations is identified so that the perturbed paths continue to remain feasible. It is shown that this can be achieved by choosing a feasible  $(R_c, d_c)$  pair from "large" regions in the  $R_c - d_c$  space. It is also highlighted that since analytical forms of the paths are known, these can be programmed in each robot, thus relaxing the need for communication, as now the robots can implement individual path tracking controllers to trace them. It is also shown that if need be, by scaling down the Pythagorean triples equally, speed and turn rate constraints that are present in hardware can be respected. Though the design is performed for a 3-robot swarm, it is also shown in this paper that additional robots can be injected at different locations on each trochoidal path, while ensuring collision avoidance with any other robot. With the introduction of these additional robots, the refresh period - the time taken by a single robot to trace the complete path, [12] - is considerably reduced; the additional robots injected in one path follow the robot that traces this path by implementing the CP.

The main contributions of the paper are as follows: the design of a CP for a 3-robot swarm that leads to them tracing closed trochoidal paths around a region of interest; the designed paths *i.* ensure geometric and speed constraints are respected; *ii.* guarantee collision avoidance; *iii.* offer flexibility in implementation from a consensus-based approach to that of control by individual robots; *iv.* allow for scaling of linear and angular speeds of the non-holonomic robots without causing any change in the resulting geometric patterns; and *v.* can be traced with an increased refresh rate by the injection of additional robots. The placement of eigenvalues The proposed designs are also implemented in an indoor mobile robot platform to demonstrate the feasibility of the design and its robustness to perturbations in initial positions of the robots.

The paper is organised as follows: In Sec. 2, the robot characteristics and their communication topology, the CP presented in [1], and the properties of the resulting trochoids are briefly described. In Sec. 3, the design of the CP

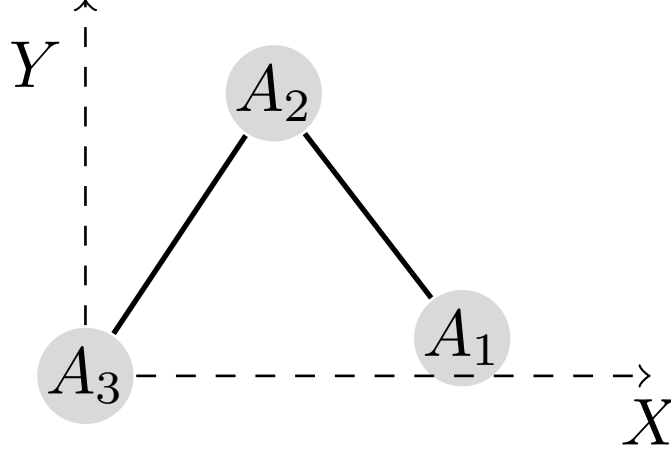


Figure 1: A connected swarm of 3 agents moving in the plane

parameters and the initial coordinates of the 3 robots is presented. In Sec. 4, simulation and experimental results of implementation are discussed. Concluding remarks are made in Sec. 5.

## 2 Preliminaries

### 2.1 Characteristics of the robots and the CP

Consider a connected swarm of 3 homogeneous agents such as shown in Fig. 1. The motion of the agents in the Cartesian  $X - Y$  plane is defined by the dynamics  $\dot{x}_i = u_{xi}$ ,  $\dot{y}_i = u_{yi}$ ,  $i = 1, 2, 3$ , where  $(x_i, y_i)$  are the coordinates of agent  $A_i$  in the Cartesian plane and  $u_{xi}, u_{yi}$  are the inputs to  $A_i$ . As discussed in [1], the CP

$$\mathbf{u} = ((\mathbf{BL}) \otimes \mathbf{S}) \mathbf{x}, \quad \mathbf{u} = [u_{x1} \ u_{y1} \ \cdots]^T, \quad \mathbf{x} = [x_1 \ y_1 \ \cdots]^T, \quad (1)$$

leads to the agents tracing trochoidal paths on the  $X - Y$  plane; these paths have the analytical expressions

$$\begin{aligned} \mathbf{x}(t) &= \mathbf{V}_L (\cos(\mathbf{J}t) \otimes \mathbf{I}_2 + \sin(\mathbf{J}t) \otimes \mathbf{S}) \mathbf{V}_R \mathbf{x}_0, \\ \mathbf{V}_L &= \mathbf{V} \otimes \mathbf{I}_2, \quad \mathbf{V}_R = \mathbf{V}^{-1} \otimes \mathbf{I}_2. \end{aligned} \quad (2)$$

In the CP (1) and (2),  $\mathbf{B} \in \mathbb{R}^{3 \times 3}$  is a user-defined diagonal matrix,  $\mathbf{L} \in \mathbb{R}^{3 \times 3}$  is the graph Laplacian,  $\mathbf{S}$  is the 2-dimensional skew-symmetric matrix;  $\mathbf{x}_0$  are the agents' initial coordinates;  $(\mathbf{V}, \mathbf{J})$  are the eigenvector and eigenvalue matrices, respectively, of  $(\mathbf{BL})$ ; and  $\mathbf{I}_2$  is the 2-D identity matrix. The operator  $\otimes$  denotes the Kronecker product.

Suppose the agents are differential drive robots with non-holonomic constraints, then these can be modeled as unicycles in the form

$$\dot{x}_i = V_i \cos \gamma_i, \quad \dot{y}_i = V_i \sin \gamma_i, \quad \dot{\gamma}_i = \omega_i, \quad (3)$$

where  $(x, y, \gamma)_i$  denotes the pose of robot  $A_i$ ,  $|V_i| \leq V_{\max}$  is its linear speed, and  $|\omega_i| \leq \omega_{\max}$  is the angular speed. Now, the CP (1) can be implemented by calculating the linear speed  $V_i = (u_{xi}^2 + u_{yi}^2)^{0.5}$ , the desired orientation  $\gamma_{i\text{Ref}} = \tan^{-1}(u_{yi}/u_{xi})$ , and the angular speed as the proportional controller  $\omega_i = K_P (\gamma_{i\text{Ref}} - \gamma_i)$  with gain  $K_P > 0$ ; note that  $\omega_i$  can also be designed as a proportional-integral controller.

### 2.2 Trochoids' properties

The type of trochoid is defined by the ratio of the largest and smallest eigenvalues of  $(\mathbf{BL})$ ; note that one of the eigenvalues is always zero. Now, by defining  $\mathbf{B} = \text{diag}([\beta_1 \ \beta_2 \ \beta_3])$ , the other two eigenvalues are given by

$$\lambda_{1,2} = \frac{\beta_1 + 2\beta_2 + \beta_3}{2} \pm \frac{1}{2} \left( (\beta_1 - \beta_3)^2 + (2\beta_2)^2 \right)^{0.5}. \quad (4)$$

It is the 3 scalars  $\beta_i$  that form the key design variables of the CP. Suppose these are chosen such that the two eigenvalues, denoted by  $\lambda_{\max} > \lambda_{\min}$ , are distinct. Now, if the ratio  $\frac{\lambda_{\max}}{\lambda_{\min}}$  is an *integer*, then the trochoidal patterns are closed.

Further, by introducing the integer design variable  $k \geq 2$ , if  $\frac{\lambda_{\max}}{\lambda_{\min}} = +(k+1)$ , then the path becomes an epitrochoid; if  $\frac{\lambda_{\max}}{\lambda_{\min}} = -(k-1)$ , then the path becomes a hypotrochoid. It is clear that the integer  $k$  represents the number of cusps in the corresponding trochoid.

In terms of these eigenvalues, the expressions (2) for agent  $A_i$  have the form

$$x_i(t) = c_{ir} \cos(\lambda_{\min}t + \phi_r) + c_{id} \cos(\lambda_{\max}t + \phi_d) + c_{i0} \cos \phi_0 \quad (5a)$$

$$y_i(t) = c_{ir} \sin(\lambda_{\min}t + \phi_r) + c_{id} \sin(\lambda_{\max}t + \phi_d) + c_{i0} \sin \phi_0, \quad (5b)$$

where the coefficients  $c_{ij}$  and the phase angles  $\phi_j$ ,  $j = r, d, 0$ , are given by

$$c_{ij} = \mathbf{V}_{i,p} (q_x^2 + q_y^2)^{0.5}, \quad \phi_j = \tan^{-1} \left( \frac{q_y}{q_x} \right) \quad (6)$$

$$q_x = [\mathbf{V}^{-1}_{p,1} \quad \mathbf{V}^{-1}_{p,2} \quad \mathbf{V}^{-1}_{p,3}] \begin{bmatrix} x_{10} \\ x_{20} \\ x_{30} \end{bmatrix}, \quad q_y = [\mathbf{V}^{-1}_{p,1} \quad \mathbf{V}^{-1}_{p,2} \quad \mathbf{V}^{-1}_{p,3}] \begin{bmatrix} y_{10} \\ y_{20} \\ y_{30} \end{bmatrix}$$

The indices  $j$  and  $p$  are given by

$$p = \begin{cases} \text{index of } \lambda_{\min} \text{ of } \mathbf{J} & \text{for } j = r \\ \text{index of } \lambda_{\max} \text{ of } \mathbf{J} & \text{for } j = d \\ \text{index of the } \mathbf{zero} \text{ eigenvalue of } \mathbf{J} & \text{for } j = 0 \end{cases} \quad (7)$$

Note that as  $\mathbf{J}$  is the diagonal matrix of eigenvalues, the index of any eigenvalue is given by its location  $(p, p)$ .

The eigenvector corresponding to the zero eigenvalue of  $(\mathbf{BL})$  consists of identical elements. Hence, the coefficients  $c_{i0} = c_{j0} = c_0$ ,  $i, j = 1, 2, 3$ ,  $i \neq j$ . This implies that all agents trace trochoidal patterns about the point with coordinates  $(c_0 \cos \phi_0, c_0 \sin \phi_0)$  as the centre. Now, by placing the agents at the ‘‘new’’ initial positions  $x'_{i0} = x_{i0} - c_0 \cos \phi_0$  and  $y'_{i0} = y_{i0} - c_0 \sin \phi_0$ , the CoR can be made to be the origin of the  $X - Y$  plane.

With this background, the main problems that are addressed in this paper are: *Design the  $i$ . elements of the matrix  $\mathbf{B}$ ,  $ii$ . the number of cusps  $k$ , and  $iii$ . the initial positions, to obtain closed and periodic epi- or hypotrochoidal paths, such that*

1. the closest, and farthest, distance to the origin of any path is larger, or smaller, than a user-defined metric;
2.  $\forall t \geq 0$ , the robots are separated by some distance, say as defined by the size of the agents; and
3. the speed and turn rate constraints are not violated.

These objectives ensure that the robots do not cross into the guarded region, in such applications, and trace large arcs that may impose communication challenges if they are equipped with limited range sensors, and also that they do not collide with each other while tracking the paths.

### 3 Design of the Trochoids

The design of trochoids is demonstrated for an epitrochoid; hypotrochoids can be designed similarly. As mentioned in the Introduction, to show that designing 10 variables - the parameters of the CP,  $\beta_{1,2,3}$ , the number of cusps  $k$ , and the initial coordinates  $\mathbf{x}_0$  - can be reduced to designing just 4, the coefficients  $c_{ir}, c_{id}$  in (5) are analysed. It can be shown that these coefficients can be expressed in the form

$$c_{ir} = \alpha_{ir} R_c, \quad c_{id} = \alpha_{id} d_c, \quad (8)$$

where  $\alpha_{ir}$  and  $\alpha_{id}$  are controlled solely by  $\mathbf{B}$  while  $R_c$  and  $d_c$  are influenced by  $\mathbf{B}$  as well as  $\mathbf{x}_0$ . These are given by

$$\alpha_{1r} = \frac{1}{\beta_d} \frac{\beta_1}{\beta_2} (\lambda_{\min} - \beta_2 - \beta_3), \quad \alpha_{1d} = \frac{1}{\beta_d} \frac{\beta_1}{\beta_2} (\lambda_{\max} - \beta_2 - \beta_3), \quad (9)$$

$$\alpha_{2r} = \frac{\beta_3 - \lambda_{\min}}{\beta_d}, \quad \alpha_{2d} = \frac{\beta_3 - \lambda_{\max}}{\beta_d}, \quad \alpha_{3r} = \alpha_{3d} = \frac{\beta_3}{\beta_d}$$

where

$$\beta_d = 4b(\beta_1\beta_2 + \beta_2\beta_3 + \beta_1\beta_3),$$

$$a = \frac{\beta_1}{2} + \beta_2 + \frac{\beta_3}{2}, \quad b = \frac{1}{2} \left( (\beta_1 - \beta_3)^2 + (2\beta_2)^2 \right)^{0.5},$$

and

$$R_c = \left( (\Gamma_R^T \mathbf{x}_{i0})^2 + (\Gamma_R^T \mathbf{y}_{i0})^2 \right)^{0.5}, \quad d_c = \left( (\Gamma_d^T \mathbf{x}_{i0})^2 + (\Gamma_d^T \mathbf{y}_{i0})^2 \right)^{0.5}, \quad (10a)$$

$$\phi_r = \tan^{-1} \left( \frac{\Gamma_{\phi_r}^T \mathbf{y}_{i0}}{\Gamma_{\phi_r}^T \mathbf{x}_{i0}} \right), \quad \phi_d = \tan^{-1} \left( \frac{\Gamma_{\phi_d}^T \mathbf{y}_{i0}}{\Gamma_{\phi_d}^T \mathbf{x}_{i0}} \right) \quad (10b)$$

where  $\mathbf{x}_{i0} = [x_{10} \ x_{20} \ x_{30}]^T$ ,  $\mathbf{y}_{i0} = [y_{10} \ y_{20} \ y_{30}]^T$ , and the coefficient vectors  $\Gamma_R, \Gamma_d, \Gamma_{\phi_r}, \Gamma_{\phi_d}$  are defined in the Appendix.

### 3.1 Design of B

For a closed epitrochoid with a user-defined number of cusps  $k$ , the ratio of the non-zero eigenvalues of  $\mathbf{BL}$ , given by (4), should also be an integer and satisfy  $\frac{\lambda_{\max}}{\lambda_{\min}} = +(k+1)$ . The parameters  $\beta_i$  are selected such that they form a Pythagorean triple, defined by  $(s_1, s_2, s_3)$ , where  $s_3^2 = s_1^2 + s_2^2$ . With this choice, these parameters are given by

$$\beta_1 = \frac{2s_3 - k(s_2 + s_1 - s_3)}{2k}, \quad \beta_2 = \frac{s_2}{2}, \quad \beta_3 = \beta_1 + s_1. \quad (11)$$

It can be seen that for a specific triple  $(s_1, s_2, s_3)$  and some  $k$ , the equality  $\beta_1 = 0$  may hold, for example, for  $k = 5$  and the triple  $(3, 4, 5)$ , including its multiples (that makes them real numbers). In such cases, from (9), it can be seen that the coefficients  $\alpha_{1r} = \alpha_{1d} = 0$ , implying that agent  $A_1$  does not move at all. This implies that for agents  $A_2$  and  $A_3$ , their CoR is agent  $A_1$ ; such choices of  $(s_1, s_2, s_3)$  and  $k$  should be avoided. Note that for hypotrochoids, the parameter  $\beta_1 = \frac{-2s_3 - k(s_2 + s_1 - s_3)}{2k}$ , while  $\beta_2, \beta_3$  are the same as in (11).

The benefit offered with the use of a Pythagorean triple, is that by scaling up, or down, the triple  $(s_1, s_2, s_3)$ , the coefficients  $c_{ir}, c_{id}$ , and hence, the magnitudes of the derivatives of  $x_i(t), y_i(t)$ , are also correspondingly scaled. This feature proves helpful in hardware implementation on robots that have bounded linear and angular speed capabilities.

### 3.2 Design of the Initial Positions

The CP (1), when expanded, reveals a form that is dependent on the *difference* in the agents' positions, rather than their absolute values. Thus, choosing the agents' initial positions to design the trochoidal trajectories can be simplified by placing one of them at the origin and finding the positions of others relative to this origin. While this approach leads to a CoR that is not the origin, once the trochoid is designed, all agents can again be shifted by known distances so that the trochoids are centered at the origin or some desired location. As a result, without any loss of generality, let agent  $A_3$  be placed at the origin; this choice makes  $(x_{30}, y_{30}) = (0, 0)$ .

To simplify further choosing initial positions, consider the distance of agent  $A_i$  from the origin, given by  $\sqrt{x_i(t)^2 + y_i(t)^2}$ . It can be shown that, for  $\alpha_{ir}, \alpha_{id} > 0$ ,  $A_i$  is closest to the origin at

$$t_{\min} = \frac{1}{k\lambda_{\min}} (2(m+1)\pi - (\phi_d - \phi_r)), \quad m = 0, 1, \dots,$$

and farthest at the time instants

$$t_{\max} = \frac{1}{k\lambda_{\min}} (2m\pi - (\phi_d - \phi_r)), \quad m = 0, 1, \dots$$

Note that if  $\alpha_{ir}, \alpha_{id}$  have different signs, then  $t_{\min}$  is when the agent is farthest and  $t_{\max}$  is when it is the closest.

It is clear from these expressions that the difference  $(\phi_d - \phi_r)$  is common to *all* agents and essentially offsets, in time, when an agent is closest to, or farthest from, the origin. A similar analysis holds for pair-wise distances between the agents; the term  $(\phi_d - \phi_r)$  shifts when a pair of agents are closest to, or farthest from, each other. Thus, without any loss of generality, this offset can be set to zero, that is, the initial positions can be selected so that  $\phi_r = \phi_d$ , in (10). With this choice, it can be shown that the agents are now co-linear and lie on a line with slope  $\tan \phi_r$ . Further, since the actual magnitudes of  $\phi_r, \phi_d$  do not influence the epitrochoidal paths, for simplicity, they can be set to zero. From (10b), this implies that the agents can all be selected to lie on the  $X$ -axis of the global Cartesian plane; note that this is a special case of the line on which the agents are co-linear. Thus, for the 3-agent swarm, the  $x$ -coordinates of agents 1 and 2,  $x_{10}$  and  $x_{20}$ , respectively, become the design variables.

To satisfy the stated geometrical constraints, which are in essence distance measures, from (10), the design of  $x_{10,20}$ , is transformed to finding the appropriate values of  $R_c, d_c$ . This is possible since the coefficients  $\Gamma_R, \dots$  have already been found, implying that  $\alpha_{ir}, \alpha_{id}$  are also known. Thus, the distance constraints can be satisfied by choosing  $R_c, d_c$ .

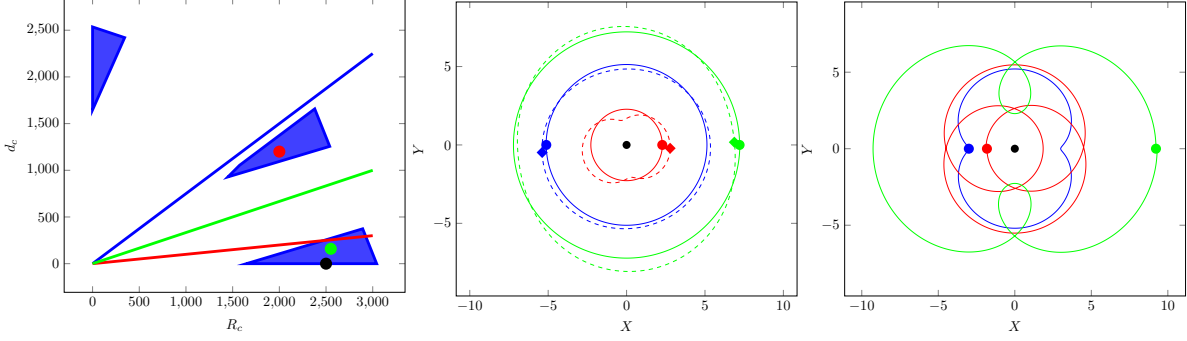


Figure 2: Design parameters  $k = 2$ ,  $s_{1,2,3} = (5, 12, 13)$ , and  $d_{0 \min} = 1.5$ ,  $d_{0 \max} = d_{CR} = 15$ , and  $d_{CT} = 0.5$ . *Left*: Feasible regions in the  $R_c - d_c$  space; *Center*: Paths for  $R_c = 2500, d_c = 0$ ; *Right*: Paths for  $R_c = 2000, d_c = 1200$ . Blue is Agent  $A_1$ , red is Agent  $A_2$ , and green is Agent  $A_3$ . The straight lines shown in the left sub-figure represent  $R_c, d_c$  values that lead to trochoidal paths that have cusps, which are locations where the angular speed of a non-holonomic robot is undefined.

### 3.3 Design of $R_c, d_c$

The parameters  $R_c, d_c$  are determined by points in the  $R_c - d_c$  space that are bounded by a set of inequalities. These inequalities capture the constraints that

1. the closest distance between the origin and any agent, denoted by  $d_{i0 \min}$ , satisfies  $d_{i0 \min} \geq d_{0 \min} > 0$ ;
2. the farthest distance from the origin to any agent, denoted by  $d_{i0 \max}$ , satisfies  $d_{i0 \max} \leq d_{0 \max}$ ;
3. the closest distance between any pair of agents,  $A_i, A_j$ , given by  $d_{ij \min}$ , satisfies  $d_{ij \min} \geq d_{CT} > 0$ ; and
4. the farthest distance between any pair of agents,  $A_i, A_j$ , given by  $d_{ij \max}$ , satisfies  $d_{ij \max} \leq d_{CR}$ .

The distance  $d_{CT}$  can be interpreted as a collision threshold, which should not be crossed by a pair of agents, while the distance  $d_{CR}$  can be interpreted as a limit on the communication range between a pair of agents. The value for  $d_{CT}$  can be selected by modeling the agents as being bounded in size by a circle of radius  $R_{Rob}$ , thus  $d_{CT} > 2R_{Rob}$ . Similarly, the distance  $d_{0 \min}$  can also be selected by modeling the asset, around which the agents traverse, as a circle of radius, say  $R_{Asset}$ , thus,  $d_{0 \min} > R_{Asset}$ . Also note that if the communication range  $d_{CR}$  is known, then the distance  $d_{0 \max}$  can be approximated as  $d_{0 \max} \approx 0.5d_{CR}$ .

In terms of  $R_c, d_c$ , for agent  $A_i$ ,  $i = 1, 2, 3$ , it can be shown that these constraints are defined by the inequalities that are linear in  $R_c, d_c$  and given by

$$|\alpha_{ir}|R_c - |\alpha_{id}|d_c \geq d_{0 \min} \quad (12a)$$

$$|\alpha_{ir}|R_c + |\alpha_{id}|d_c \leq d_{0 \max} \quad (12b)$$

$$|\alpha_{ir} - \alpha_{jr}|R_c - |\alpha_{id} - \alpha_{jd}|d_c \geq d_{CT}, i \neq j \quad (12c)$$

$$|\alpha_{ir} - \alpha_{jr}|R_c + |\alpha_{id} - \alpha_{jd}|d_c \leq d_{CR}, i \neq j \quad (12d)$$

$$R_c \geq 0, d_c \geq 0 \quad (12e)$$

For the 3-agent swarm, from a geometrical perspective, the values of  $\mathbf{B}$  and  $(R_c, d_c)$  that satisfy these 14 inequalities lead to a feasible set of epitrochoidal paths. As is evident, these inequalities form polyhedra in the  $R_c - d_c$  space.

*Example:* Choosing  $k = 2$ , the Pythagorean triple  $(5, 12, 13)$ , and the distances  $d_{0 \min} = 1.5$ ,  $d_{0 \max} = d_{CR} = 15$ , and  $d_{CT} = 0.5$  leads to the formation of the regions shown in the left sub-figure of Fig. 2. Selection of a particular  $(R_c, d_c)$  pair from these regions determines the path. For example, for the solution  $R_c = 2500, d_c = 0$  (marked by the black circle in the left sub-figure in Fig. 2), the paths become circles (denoted by the solid lines in the center sub-figure in Fig. 2), while for  $R_c = 2000, d_c = 1200$  (marked by the red circle in the left sub-figure in Fig. 2), the paths are epitrochoids (also the solid lines in the right sub-figure in Fig. 2). For a feasible  $(R_c, d_c)$  pair, the initial coordinates  $x_{10}, x_{20}$  are obtained from (10a); note that, by construction,  $x_{30} = 0$  and  $y_{i0} = 0 \forall i$ . In both the figures in Fig. 2, the agents' initial positions are recalculated to shift the CoR to the origin - as can be seen, this recalculation shifts only the  $x$ -coordinates of the initial positions.

### 3.4 Injecting Additional Agents

In this section, conditions are derived when additional agents can be injected to the 3 epitrochoidal paths designed following the described procedure. If agent  $A_i$ ,  $i = 1, 2, 3$ , traces path  $P_i$ , then additional agents  $A_{mi}$ ,  $m = 1, \dots$ , are injected to path  $P_i$ . It is clear that agents  $A_{mi}$  should be placed such that they do not collide with other agents  $A_j$ ,  $j \neq i$  and those injected on paths  $P_j$ ; note that agents on the same path, if injected at different locations, naturally do not collide with each other. Consider two epitrochoids in the parametric form

$$x_{mi} = (k+1)r_i c(\theta+\phi) - d_i c((k+1)(\theta+\phi)) \quad (13a)$$

$$y_{mi} = (k+1)r_i s(\theta+\phi) - d_i s((k+1)(\theta+\phi)) \quad (13b)$$

and

$$x_j = (k+1)r_j c(\theta) - d_j c((k+1)\theta) \quad (13c)$$

$$y_j = (k+1)r_j s(\theta) - d_j s((k+1)\theta), \quad (13d)$$

where  $(r_i, d_i)$  and  $(r_j, d_j)$  are the parameters of paths  $P_i$  and  $P_j$ , respectively. In (13a) and (13b), the angle  $\phi$  defines the location of agent  $A_{mi}$  that is offset from agent  $A_i$  on the path  $P_i$ .

Thus, for agents  $A_{mi}$  and  $A_j$  to avoid collisions, a suitable  $\phi$  should be found. To avoid collisions, the squared separation distance between  $A_{mi}$  and  $A_j$ , given by  $d_{mij}^2 = (x_{mi} - x_j)^2 + (y_{mi} - y_j)^2$ , should satisfy  $d_{mij}^2 \geq d_{CT}^2$ , where  $d_{CT}$  is the collision avoidance threshold. On expanding, it follows that  $d_{mij}^2$  is a non-linear function of  $\phi$  and hence to find the infeasible values of  $\phi$ , which are the solutions to the minimisation problem  $d_{mij}^2 - d_{CT}^2 = 0$ , numerical routines may have to be adopted. However, for some specific values of  $\phi$ , a closed-form solution for the *feasible* solutions can be obtained; this procedure is now discussed.

Choose  $\phi = \frac{\pi}{2}$  and let  $k = 2n$ ,  $n = 1, 3, 5, \dots$ . For this case, the difference  $\delta_{mij}^2 = (d_{mij}^2 - d_{CT}^2)$  has the expression

$$(k+1)^2 (r_i^2 + r_j^2) + (d_i^2 + d_j^2) + 2(k+1)(r_i d_i - r_j d_j) \cos(k\theta) + 2(k+1)(r_i d_j + r_j d_i) \sin(k\theta) - d_{CT}^2 \quad (14)$$

and the values of  $\theta$  when this difference has a minimum/maximum are given by  $k\theta = \tan^{-1}\left(\frac{r_i d_j + r_j d_i}{r_i d_i - r_j d_j}\right)$  and  $k\theta = \tan^{-1}\left(\frac{r_i d_j + r_j d_i}{r_i d_i - r_j d_j}\right) + \pi$ . Substituting the first of these solutions in (14) results in the difference  $\delta_{mij}^2$  achieving its minimum value, which is given by

$$\delta_{mij \min Ke}^2 = \left( (k+1)(r_i^2 + r_j^2)^{0.5} - (d_i^2 + d_j^2)^{0.5} \right)^2 - d_{CT}^2. \quad (15)$$

Thus, if the trochoidal parameters,  $r_{i,j}$ ,  $d_{i,j}$ , and  $k$  are such that  $\delta_{mij \min Ke}^2 \geq 0$ , then agent  $A_{mi}$  can be injected in the path of agent  $A_i$ , but offset by an angle  $\phi = \frac{\pi}{2}$ , so that it does not collide with agent  $A_j$ . For the case when  $k = 2n$ , where  $n = 2, 4, \dots$ , the result (15) also holds, but achieves its minimum at the second solution of  $k\theta$ . When  $k$  is odd, the minimum of the difference  $\delta_{mij}^2$  is given by

$$\delta_{mij \min Ko}^2 = \left( (k+1)(r_i^2 + r_j^2)^{0.5} - (d_i + d_j) \right)^2 - d_{CT}^2. \quad (16)$$

In this case, the trochoidal parameters,  $r_{i,j}$ ,  $d_{i,j}$ , and  $k$  should be selected such that the inequality  $\delta_{mij \min Ko}^2 \geq 0$  holds. It can be shown that the inequalities  $\delta_{mij \min Ke}^2 \geq 0$  and  $\delta_{mij \min Ko}^2 \geq 0$  should be satisfied for  $\phi = \frac{3\pi}{2}$  as well.

For the case  $\phi = \pi$ , the expressions for the minimum of  $\delta_{mij}^2$  is dependent on the signs of the products  $(r_i + r_j)(d_i + d_j)$  and  $(r_i + r_j)(d_j - d_i)$ . It is remarked that, from the definitions of the coefficients in (8) and comparing them with the coefficients of the parametric forms in (13), the signs of these products are the same as that of the products  $(\alpha_{ir} + \alpha_{jr})(\alpha_{id} + \alpha_{jd})$  and  $(\alpha_{ir} + \alpha_{jr})(\alpha_{jd} - \alpha_{id})$ , respectively. Thus, the following conditions can be derived for  $\phi = \pi$ :

1. For  $k$  even and  $(\alpha_{ir} + \alpha_{jr})(\alpha_{id} + \alpha_{jd}) > 0$ , the minimum distance,  $\delta_{mij \min Ke}^2 = ((k+1)(r_i + r_j) - (d_i + d_j))^2 - d_{CT}^2$ ;
2. For  $k$  even and  $(\alpha_{ir} + \alpha_{jr})(\alpha_{id} + \alpha_{jd}) < 0$ ,  $\delta_{mij \min Ke}^2 = ((k+1)(r_i + r_j) + (d_i + d_j))^2 - d_{CT}^2$ ;

3. For  $k$  odd and  $(\alpha_{ir} + \alpha_{jr})(\alpha_{id} - \alpha_{jd}) > 0$ ,  $\delta_{mij \min Ko}^2 = ((k+1)(r_i + r_j) - (d_j - d_i))^2 - d_{CT}^2$ ;
4. For  $k$  odd and  $(\alpha_{ir} + \alpha_{jr})(\alpha_{id} - \alpha_{jd}) < 0$ ,  $\delta_{mij \min Ko}^2 = ((k+1)(r_i + r_j) + (d_j - d_i))^2 - d_{CT}^2$ ;

Thus, by considering the constraints that should hold for  $\phi = \frac{\pi}{2}, \pi$ , and  $\frac{3\pi}{2}$ , 3 additional agents can be injected in each path, increasing the number of agents to 12; the number of constraints also increases by 18 (6 constraints for the 3 values of  $\phi$  per path, that has to be checked for 3 paths).

For the example considered in Sec. 3.3, the variation in the feasible regions in the  $R_c - d_c$  space with the inclusion of 3 additional agents per path, and the paths traced by them, are shown in Fig. 3. As can be seen, the region in the center of the  $R_c - d_c$  space has significantly shrunk, while that close to the  $R_c$ -axis has not. This implies that for epitrochoidal paths that are close to circles, as shown in Fig. 2, additional agents can be injected quite easily - this is a natural result as none of these paths intersects each other, while those found from the center of the  $R_c - d_c$  space self-intersect and with other paths as well. It is remarked that it may be possible to inject more than 3 agents per path, by solving the minimisation problem numerically for feasible  $\phi$ . As should be evident, the frequency with which a point in the 2-D space is visited increases linearly as the number of agents, thus increasing the refresh rate.

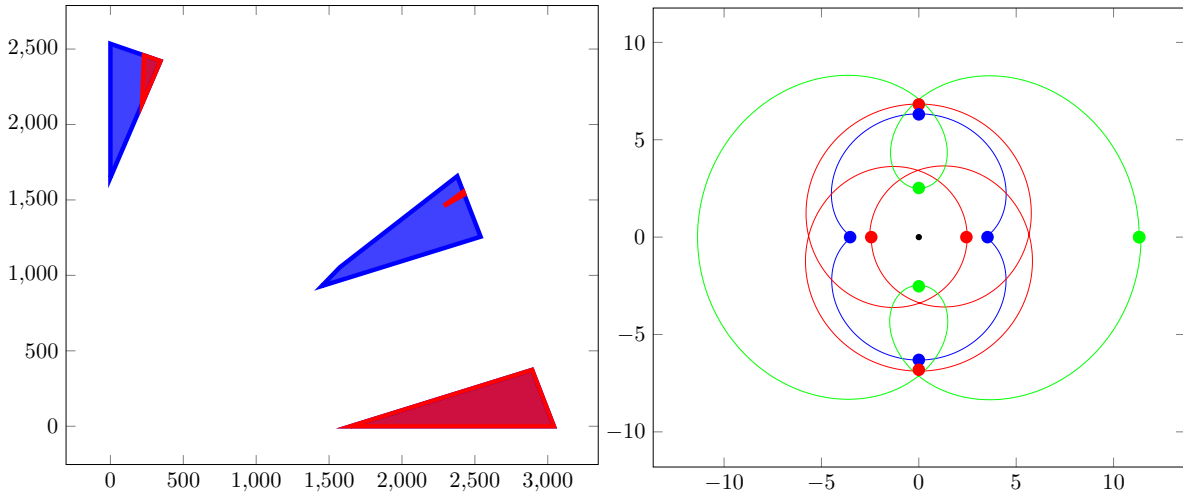


Figure 3: *Left*: Reduction in the feasible regions in the  $R_c - d_c$  space (marked in red) for the example considered in Sec. 3.3; *Right*: Initial positions and paths traced by the 12 agents for the solution  $R_c = 2400, d_c = 1525$ .

## 4 Results

Prior to presenting experimental results, challenges that can arise in implementation of the proposed design are identified and addressed analytically. Mainly, the issues of implementing the designed CP on non-holonomic robots with input bounds and perturbations in the designed initial positions are considered.

### 4.1 Inclusion of Speed and Turn-rate Constraints

As introduced in Sec. 2.1, the non-holonomic robots whose kinematics are defined in (3) have bounded linear speed and turn-rate, defined by the constraints  $|V| \leq V_{\max}$  and  $|\omega| \leq \omega_{\max}$ , respectively. In the robot platform used for experimental evaluation, it is  $V, \omega$  that should be provided as inputs; thus, considering their limits is crucial. In this section, the influence of such constraints on feasible regions in the  $R_c - d_c$  space is discussed.

For analysis, the parametric form of the epitrochoid, (13c) and (13d), are used. In terms of the parameter  $\theta$ , the linear speed and turn-rate are given by

$$V(\theta) = \pm \left( \left( \frac{dx}{d\theta} \right)^2 + \left( \frac{dy}{d\theta} \right)^2 \right)^{0.5} = \pm (k+1) (r^2 + d^2 - 2rd \cos(k\theta))^{0.5}, \quad (17a)$$



and

$$\omega(\theta) = \frac{\frac{d^2 y}{d\theta^2} \frac{dx}{d\theta} - \frac{d^2 x}{d\theta^2} \frac{dy}{d\theta}}{V(\theta)^2} = \frac{r^2 + d^2(k+1) - rd(k+2)\cos(k\theta)}{r^2 + d^2 - 2rd\cos(k\theta)}. \quad (17b)$$

If the coefficients satisfy  $d = r$ , which is a cusp of the epitrochoid, then, at  $\theta = \frac{2m\pi}{k}$ ,  $m = 0, 1, \dots$ , the linear speed  $V(\theta = \frac{2m\pi}{k}) = 0$  and the turn-rate  $\omega(\theta = \frac{2m\pi}{k})$  becomes undefined. This is because, at the cusp, the robot would have to come to a stop, change its orientation, and trace the path from this new orientation. To avoid this discontinuity, if still achievable behaviour, certain values of  $(R_c, d_c)$  are eliminated. These values correspond to when the coefficients  $c_{ir}$  and  $c_{id}$ , defined in (5) and (8), become  $c_{ir} = c_{id} \forall i$ . As an added safety margin, these infeasible regions are expanded using the inequality

$$(1 - \epsilon) \leq \left| \frac{c_{id}}{c_{ir}} \right| \leq (1 + \epsilon), \quad 0 < \epsilon < 1. \quad (18)$$

Thus, the feasible regions may further shrink when this constraint is added.

At  $\theta = \frac{2(m+1)\pi}{k}$ ,  $m = 0, 1, \dots$ , the linear speed and turn-rate achieve their maximum, given by

$$V_{\theta \max} = \pm(k+1)r \left( \frac{d}{r} + 1 \right), \quad (19a)$$

$$\omega_{\theta \max} = \frac{\left( \frac{d}{r} + 1 \right)^2 + k \frac{d}{r} \left( \frac{d}{r} + 1 \right)}{\left( \frac{d}{r} + 1 \right)^2}, \quad (19b)$$

respectively. Clearly, if  $\frac{d}{r} \gg 1$ , the linear speed requirement becomes large, possibly violating the bound  $V \leq V_{\max}$ , although the angular speed demand may be low; the condition  $\frac{d}{r} \gg 1$  approximates the trajectory to a circle with radius  $d$  - a large number. These solutions of  $(R_c, d_c)$  may also need to be rejected. To summarise, during implementation on non-holonomic robots with bounds on their inputs, the feasible regions in the  $R_c - d_c$  shrink further, as those solutions are eliminated that: *i.* lead to cusps even for one of the agents; and *ii.* result in high demands on the linear or angular speeds.

In the example presented in Sec. 3.3, the values of  $R_c, d_c$  that lead to cusps for each agent are shown by the straight lines (by setting  $\epsilon = 0$  in (18)) in the left sub-figure in Fig. 2. As can be seen, for Agent  $A_2$  (red line), there are parts of the feasible region that lead to a discontinuity in turn-rate. The almost circular paths in the center sub-figure demand constant turn-rate and hence zero angular acceleration, while for those in the right sub-figure, larger values of turn-rate and angular acceleration are expected at the troughs of the trochoidal path. Thus, a trade-off needs to be performed in the selection of these paths. A decision variable that can be employed is the area traced by the robots moving on these paths, when they are performing a surveillance or guarding task - a larger surface area coverage can be expected with trochoidal paths that contain troughs/crests than paths that resemble circles.

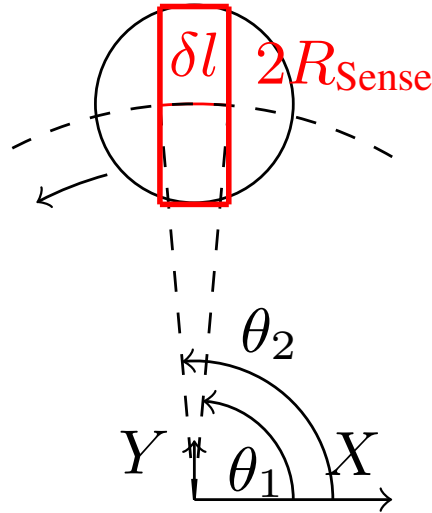


Figure 4: Computing the area covered by a robot with sensing radius  $R_{\text{Sense}}$  tracing a path

*Area Coverage:* To design the paths, in Sec. 3.3, an agent is modeled as being bounded in size by a circle of radius  $R_{\text{Rob}}$ . Further, assume that each agent is also equipped with a range sensor with range  $R_{\text{Sense}} \gg R_{\text{Rob}}$  and  $360^\circ$  sensing around the robot (for instance, with a LIDAR sensor or equivalent). Now, from the schematic shown in Fig. 4, the sensing area covered by an agent  $A_i$  in one period of the trochoidal path can be calculated. The infinitesimally small area (in red) is given by the product  $2R_{\text{Sense}}\delta l_i$ , where  $\delta l_i = \sqrt{\left(\frac{\partial x_i}{\partial \theta}\right)^2 + \left(\frac{\partial y_i}{\partial \theta}\right)^2}$  is the length of the trochoidal path (expressed in the parametric form) traced for an infinitesimally small angle  $\delta\theta = (\theta_2 - \theta_1)$ . Thus, as the trochoids are periodic and closed, the total sensing area covered by agent  $A_i$  is given by

$$\text{SenseArea}_i = 2R_{\text{Sense}} \int_0^{2\pi} \sqrt{\left(\frac{\partial x_i}{\partial \theta}\right)^2 + \left(\frac{\partial y_i}{\partial \theta}\right)^2} d\theta. \quad (20)$$

Thus, the total area covered by all 3 agents is simply the sum  $\sum_{i=1}^3 \text{SenseArea}_i$ .

From the feasible regions shown in Fig. 2, for the design pair  $(R_c, d_c) = (2500, 0)$ , which leads to circular paths, the total sensing area is found to be 146.98 (determined numerically<sup>1</sup>), whereas for  $(R_c, d_c) = (2000, 1200)$ , which results in epitrochoids with pronounced troughs/crests, the total area is 243.84 (all in appropriate units). Thus, a considerable increase in sensing area can be achieved with the use of trochoidal paths, which may require operating close to the inputs' limits of the individual agents, than circles, which do not demand such input magnitudes.

## 4.2 Perturbing Initial Positions

In Sec. 3.2, it is shown that feasible epitrochoidal paths can be generated by choosing the agents' initial positions to lie on the  $X$ -axis, say  $(x_{i0F}, 0)$ . The effect of perturbations in these initial positions is now analysed; the other design parameters remain unchanged. Let the perturbed position of each agent be given by  $(x_{i0F} + \epsilon_i \cos \psi_i, \epsilon_i \sin \psi_i)$ , where,  $\epsilon_i > 0$ ,  $\psi_i \in [0, 2\pi] \forall i$ . These new initial positions lead to different values of  $(R_c, d_c)$ , denoted by  $(R_{cp}, d_{cp})$ , hence, different trochoidal paths, and also for the paths to evolve about a new CoR, since, from (10),  $\phi_{r,d,0} \neq 0$ , and hence, in (5),  $c_0 = c_{i0} = c_{j0} \neq 0$ .

For the example considered in Sec. 3.3, the initial positions that are found for the solution  $R_c = 2500, d_c = 0$  (black circle in the left sub-figure in Fig. 2) are perturbed (with  $\epsilon_i = 1 \forall i$  and  $\psi_i$  randomly chosen to lie in  $[0, 2\pi]$ ). These perturbed initial positions result in the  $R_c, d_c$  pair marked by the green circle in the same region, which in turn lead to different trochoidal paths (dashed lines in the center sub-figure in Fig. 2); with perturbations, the path closest to the CoR exhibits a clear epitrochoidal feature, which is different from the circular path generated in the absence of any perturbations. From this illustration, the importance of finding "large" regions in the  $R_c - d_c$  space can be appreciated, as such "large" regions will hold solutions for feasible paths even in the presence of perturbed initial positions.

It can be shown that since  $(R_{cp}, d_{cp})$  and the new CoR is common to all agents, the smallest and largest distances between any pair of agents continue to satisfy the inequalities (12c) and (12d), respectively. Thus, perturbing the initial positions of the agents does not lead to collisions between them. On the other hand, the smallest distance-to-the-origin inequality, (12a), may be violated. Bounds on the magnitudes of perturbation,  $\epsilon_i$ , are now derived so that (12a) still holds.

This analysis is performed for a closed epitrochoid with the origin as the CoR and expressed in the parametric form  $x_\theta = (k+1)rc_\theta - dc_{(k+1)\theta}$ ,  $y_\theta = (k+1)rs_\theta - ds_{(k+1)\theta}$ , where  $c_{(\cdot)} = \cos(\cdot)$ ,  $s_{(\cdot)} = \sin(\cdot)$ . A point on this epitrochoid is closest to the origin at  $\theta = 0, \pi$  with distance  $(r(k+1) - d)$ . Now, let the initial coordinates, evaluated at  $\theta = 0$ , be perturbed such that the CoR of the perturbed epitrochoid (PE) has the coordinates  $(\delta \cos \gamma_T, \delta \sin \gamma_T)$ ,  $\delta > 0$  and its orientation is changed by an angle  $\gamma_R$ . Thus, the parametric form of the PE becomes

$$\begin{aligned} x_{\theta P} &= (k+1)rc_{\theta+\gamma_R} - dc_{(k+1)\theta+\gamma_R} + \delta c_{\gamma_T}, \\ y_{\theta P} &= (k+1)rs_{\theta+\gamma_R} - ds_{(k+1)\theta+\gamma_R} + \delta s_{\gamma_T}. \end{aligned}$$

The squared distance of any point on the PE to the origin is given by

$$\begin{aligned} d_{eP}^2 &= (k+1)^2 r^2 + d^2 - 2(k+1)rdc_{k\theta} \\ &\quad + \delta^2 + 2\delta((k+1)rc_{(\gamma_R-\gamma_T+\theta)} - dc_{(\gamma_R-\gamma_T+(k+1)\theta)}). \end{aligned}$$

<sup>1</sup><https://in.mathworks.com/help/matlab/ref/integral.html>

As the magnitudes of the perturbation is expected to be small, its influence on  $d_{eP}^2$  is analysed by linearising this expression about the following points: *i.*  $\delta = \gamma_{R,T} = \theta = 0$  and *ii.*  $\delta = \gamma_{R,T} = 0$  and  $\theta = \pi$ . It can be seen that these are the points when the original epitrochoid is closest to the origin. It can be shown that in the vicinity of  $\theta = 0$ ,  $d_{eP}^2 \leq (r(k+1) - d + \delta)^2$ , while in the vicinity of  $\theta = \pi$ ,  $d_{eP}^2 \leq (r(k+1) - d - \delta)^2$ . Thus, the PE shifts closer to the origin at  $\theta = \pi$  and farther away at  $\theta = 0$ ; in both cases, by no more than  $\delta$ . This shift in the paths can be observed in the dashed lines in the center sub-figure in Fig. 2.

Now, the point at  $\theta = 0$  on the PE has the coordinates  $[(r(k+1) - d)c_{\gamma_R} + \delta c_{\gamma_T}], [(r(k+1) - d)s_{\gamma_R} + \delta s_{\gamma_T}]$ . Therefore, for small  $\gamma_{R,T}$ , these coordinates can be approximated to  $[(r(k+1) - d + \delta), (r(k+1) - d)\gamma_R]$ . Thus, the magnitude of perturbation  $\epsilon_i \approx \left(\delta^2 + (r(k+1) - d)^2 \gamma_R^2\right)^{0.5}$ ; the angle  $\psi_i \approx \frac{(r(k+1) - d)\gamma_R}{\delta}$ . The implication of perturbed initial conditions is that the agents can enter the guarded region around the asset, thus violating one of the key design requirements. To avoid this issue, the inequality (12a) is amended to

$$d_{i0 \min} = \left| |\alpha_{ir}|R_c - |\alpha_{id}|d_c \right| \geq (d_{0 \min} + \delta), \quad (21)$$

so that perturbations of magnitude  $\epsilon_i$  in the initial positions do not lead to any constraint violations. While the farthest distance inequality (12b) should be similarly modified, if  $d_{i0 \max} \gg \delta$ , then, these perturbations do not affect this constraint greatly.

### 4.3 Hardware Implementation

The proposed algorithms are implemented on the QBOT 2E mobile robot platform by Quanser<sup>2</sup>. All experiments are conducted in the Autonomous Vehicles Research Studio<sup>3</sup>, also from Quanser, using 3 robots. The Studio allows for inter-robot communication, with which robots exchange their coordinates with their neighbours and hence the CP, (1), can be implemented; the communication topology as shown in Fig. 1 is adopted. The robots' coordinates, including orientation,  $\gamma_i$ , are measured by a tracking system built-in to the Studio. To control the motion of the robots, their linear and angular velocities need to be provided. The angular velocity is implemented as the proportional-integral controller with feed-forward term, given by

$$\omega_i = K_P (\gamma_{i\text{Ref}} - \gamma_i) + K_I \int_0^t (\gamma_{i\text{Ref}} - \gamma_i) dt + \left( \frac{d\gamma_{i\text{Ref}}}{dt} \right),$$

where  $K_P > 0$  and  $K_I > 0$  are the gains; the linear speeds, reference orientation angles and their time-derivatives are determined using the formulae presented in Sec. 2.1.

The robots move in an environment which is a square of size  $2 \times 2 \text{ m}^2$ ; the robots themselves are circles with radius,  $R_{\text{Rob}} = 0.015 \text{ m}$ . The distances used in the inequalities (12) are set as  $d_{0 \min} = 0.01$ ,  $d_{0 \max} = 1.8$ ,  $d_{cR} = 4$ , and  $d_{cT} = 0.5$  (all in m). Owing to the relatively large size of the robots with respect to the environment and to obtain "reasonable" paths, it became necessary to choose  $d_{0 \min} = 0.01$ . Similar to the results shown in Fig. 2, two classes of paths are generated which the robots trace: *i.* circular and *ii.* epitrochoidal. The reference (dashed lines) and actual paths (solid lines) in the  $X - Y$  plane for these two cases are shown in Figs. 5 and 6, respectively. The evolution in time of the distances of the three agents from the origin as well as inter-robot pair-wise distances are shown for the epitrochoidal path in Fig. 7. As can be seen, all distance constraints listed in (12) are satisfied by the agents. Thus, no pair of agents collide with each other and neither do they trace large paths that would cause them to cross the boundaries of the environment.

The design variables are presented in Table 1. As mentioned in the Introduction, the proposed CP design procedure simplifies the selection of 10 variables to just 4. To generate these results, the robots were driven to their respective initial positions prior to the implementation of the designed CP. The reference and actual initial positions of the three agents are also listed in Table 1; as these are slightly different, the paths traced by the robots also differ from the designed ones, but are still feasible. Further, the magnitudes of the Pythagorean triples, denoted by  $s_{1,2,3}$ , in Table 1 were scaled *down* by a factor of 0.01 for the circular path and by a factor of 0.0015 for the epitrochoidal path; with this scaling down of the speed, the robots took close to 6 min to trace one period of the path. As the robot platform allows for only the linear speed and turn-rate as the inputs to each robot, instead of controlling the input to each wheel of the robot, this scaling down had to be performed to ensure the robots trace the designed paths; the time-histories of the linear speeds and turn-rates are presented in Fig. 8. Note that the analysis in Sec. 4.1 focussed on ensuring that paths with cusps are avoided; the upper-limits on speed and turn-rate constraints are dependent on the robot platform.

As can be seen in the left sub-figure in Fig. 5, for the chosen Pythagorean triple of  $s_{1,2,3} = (5, 12, 13)$ , the region in the  $R_c - d_c$  space that would yield a non-circular path is much smaller than the other two regions; as discussed in

<sup>2</sup><https://www.quanser.com/products/qbot-2e/>

<sup>3</sup><https://www.quanser.com/products/autonomous-vehicles-research-studio/>

Table 1: Parameters used to generate the paths shown in Figs. 5 and 6.

Path	Design Variables	Reference and Actual (in []) initial positions
Circular (Fig. 5)	$s_{1,2,3} = (5, 12, 13)$	$(-1.024, 0)$ $[-1.014; 0.001]$ $(0.455; 0)$ $[0.447; 0.001]$ $(1.441; 0)$ $[1.442; 0.001]$
Epitrochoidal (Fig. 6)	$s_{1,2,3} = (7, 24, 25)$	$(-0.16, 0)$ $[-0.161; 0.004]$ $(-0.949; 0)$ $[-0.94; 0.001]$ $(1.584; 0)$ $[1.588; 0.001]$
$k = 2, d_{0 \min} = 0.01, d_{0 \max} = 1.8, d_{CR} = 4, \text{ and } d_{CT} = 0.5$		

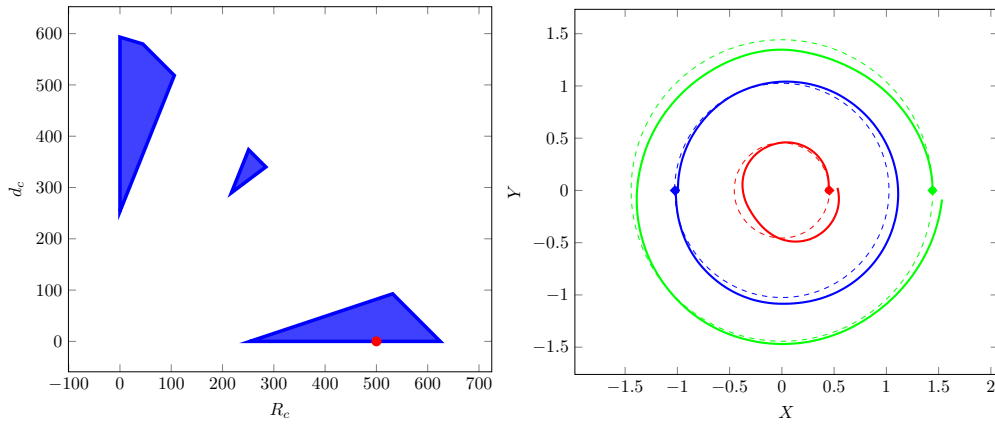


Figure 5: Design parameters:  $k = 2, s_{1,2,3} = (5, 12, 13), d_{0 \min} = 0.01, d_{0 \max} = 1.8, d_{CR} = 4, \text{ and } d_{CT} = 0.5$ . *Left*: Feasible regions in the  $R_c - d_c$  space; *Right*: Reference (dashed) and Actual (solid) Paths for  $R_c = 500, d_c = 0$ . Blue is Agent  $A_1$ , red is Agent  $A_2$ , and green is Agent  $A_3$ .

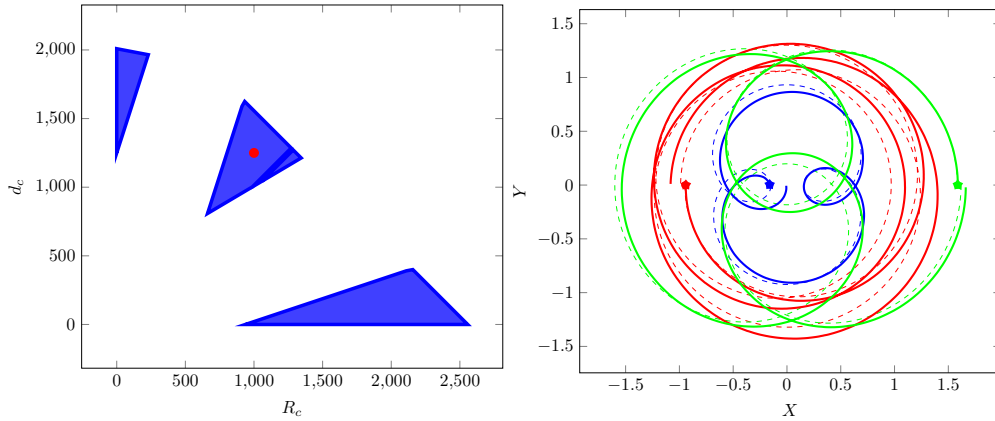


Figure 6: Design parameters:  $k = 2, s_{1,2,3} = (7, 24, 25), d_{0 \min} = 0.01, d_{0 \max} = 1.8, d_{CR} = 4, \text{ and } d_{CT} = 0.5$ . *Left*: Feasible regions in the  $R_c - d_c$  space; *Right*: Reference (dashed) and Actual (solid) Paths for  $R_c = 1000, d_c = 1250$ . Blue is Agent  $A_1$ , red is Agent  $A_2$ , and green is Agent  $A_3$ .

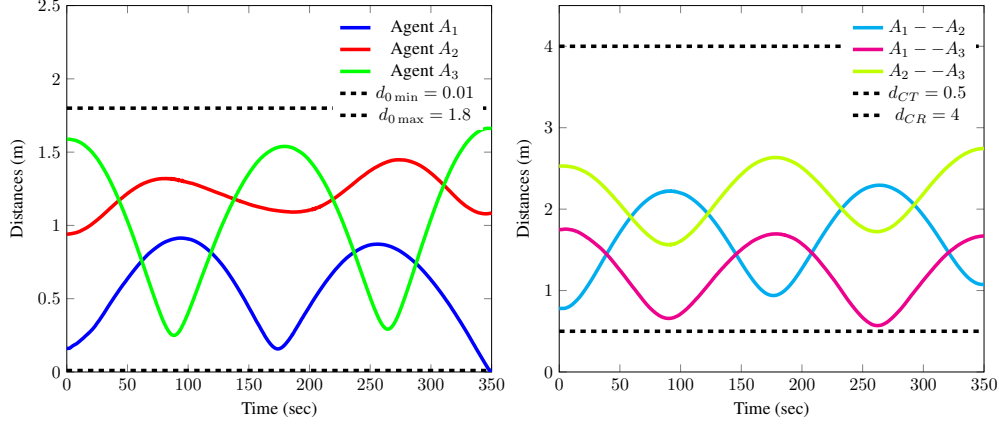


Figure 7: *Left*: Temporal variation in the distances of the robots from the origin; *Right*: Temporal variation of the pair-wise distances between the robots. The path parameters are based on the values shown in Fig. 6.

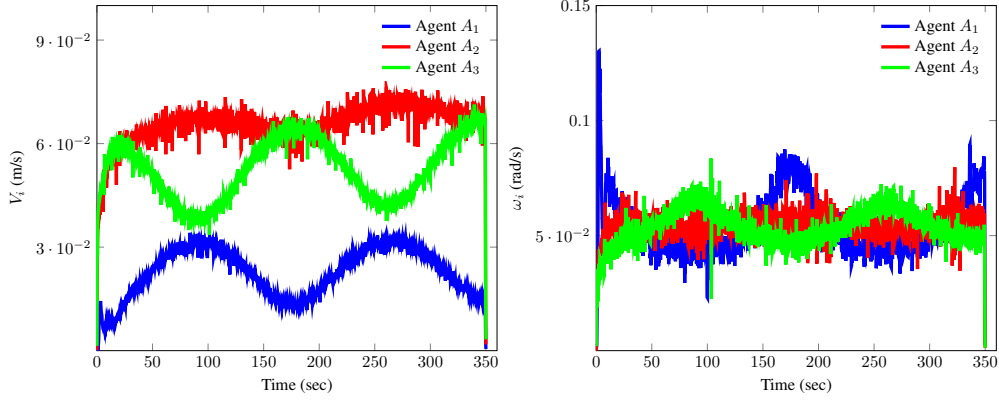


Figure 8: *Left*: Temporal variation in the linear speeds of the robots; *Right*: Temporal variation of the turn-rates the robots. The path parameters are based on the values shown in Fig. 6.

Sec. 4.2, selecting a solution for  $(R_c, d_c)$  from this “small region” would not allow for feasible paths to be traced if the initial positions are perturbed. On the other hand, from the left sub-figure in Fig. 6, as the corresponding region is larger, greater perturbations in the initial positions can be tolerated. Thus, for each class of path, the  $(R_c, d_c)$  pair is selected (red circles in both sub-figures), so that perturbations can be handled.

The results section is concluded by showing the influence of two design parameters: the Pythagorean triple and  $k$ , on the sizes of the feasible regions in the  $R_c - d_c$  space. The change in these regions for three different triples,  $\mathcal{S}_1 = \{5, 12, 13\}$ ,  $\mathcal{S}_2 = \{8, 15, 17\}$ , and  $\mathcal{S}_3 = \{12, 35, 37\}$ , while keeping  $k = 2$  as a constant, is shown in the left sub-figure in Fig. 9. Similarly, for  $\mathcal{S}_1$  and varying  $k$ , the regions are shown in the right sub-figure in Fig. 9. To make comparison easier, the  $R_c, d_c$  values for each set are normalised to lie between  $[0, 1]$ . As can be seen, both parameters play a crucial role in the generation of feasible paths. While near circular paths can be generated by all sets, regions that lead to paths that resemble trochoids reduce considerably for some combinations. As discussed before, it helps to choose regions that are “large” in order to accommodate perturbations in initial positions and those lead to larger sensing area coverage.

## 5 Conclusions

In this paper, a CP developed to enable the creation of artistic patterns is designed for a connected swarm of non-holonomic robots. By considering a swarm of 3 robots, parameters of closed and periodic trochoidal paths are calculated that ensure distance constraints are satisfied; these constraints consider robots of finite-size, hence guarantee collision avoidance if satisfied, and communication range between the robots. Further, constraints on speed and turn-rate as well as perturbations in the initial positions of the robots are also considered in the design. By exploiting the

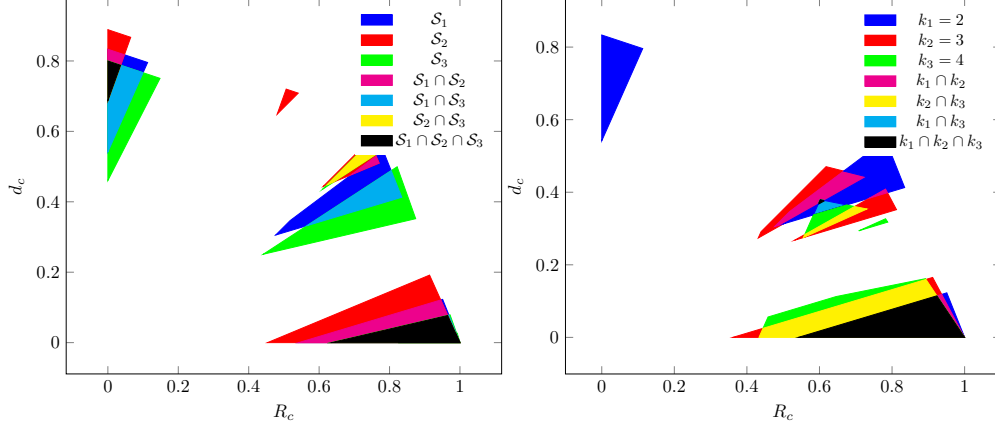


Figure 9: Regions in the (normalised)  $R_c - d_c$  space: *Left*: for the Pythagorean triples:  $\mathcal{S}_1 = \{5, 12, 13\}$ ,  $\mathcal{S}_2 = \{8, 15, 17\}$ , and  $\mathcal{S}_3 = \{12, 35, 37\}$ ;  $k = 2$  is fixed. *Right*: for different values of  $k$ :  $k_1 = 2, k_2 = 3, k_3 = 4$ ; the Pythagorean triple is fixed to  $\mathcal{S}_1$ . Distance parameters are the same as used to generate the regions in Fig. 2.

properties of the CP and the characteristics of trochoids, the proposed design procedure reduces the number of design variables and also allows for scaling temporal movements without causing any change in the geometric patterns. Conditions when additional robots may be added to the paths designed for 3 robots are also identified. The design of two types of closed paths - circular and epitrochoidal - is also implemented on a hardware platform; the trade-offs that result between these types of paths are also highlighted.

This work can be naturally extended to the 3-D case, for instance, with ground robots replaced by drones, so that additional applications can be covered; another extension is considering more than 3 agents. Alternative solutions for the design of the CP can be considered that can yield non-closed paths, which may present desirable properties in some applications. Changing these design parameters dynamically, while ensuring the identified constraints continue to be satisfied, is another extension to this work.

## APPENDIX

The coefficient vectors  $\Gamma_R = [\gamma_{r1} \ \gamma_{r2} \ \gamma_{r3}], \dots$  in (10) are given by

$$\begin{aligned}
 \gamma_{r1} &= \beta_2 (\beta_1 + 2\beta_2 + \beta_3 + 2b) \\
 \gamma_{r2} &= \beta_1 (\beta_1 - \beta_3 + 2b) - 2\beta_2\beta_3 \\
 \gamma_{r3} &= -(\beta_1^2 + 2\beta_2^2 - \beta_1\beta_3 - \beta_2\beta_3 + \beta_1\beta_2 + 2b\beta_1 + 2b\beta_2) \\
 \gamma_{d1} &= \beta_2 (\beta_1 + 2\beta_2 + \beta_3 - 2b) \\
 \gamma_{d2} &= \beta_1 (\beta_1 - \beta_3 - 2b) - 2\beta_2\beta_3 \\
 \gamma_{d3} &= -(\beta_1^2 + 2\beta_2^2 - \beta_1\beta_3 - \beta_2\beta_3 + \beta_1\beta_2 - 2b\beta_1 - 2b\beta_2) \\
 \gamma_{\phi r1} &= -\beta_2 (\beta_1 + \beta_3 + 2\beta_2 + 2b) \\
 \gamma_{\phi r2} &= -\beta_1 (\beta_1 - \beta_3 + 2b) + 2\beta_2\beta_3 \\
 \gamma_{\phi r3} &= \beta_1^2 + 2\beta_2^2 - \beta_1\beta_3 - \beta_2\beta_3 + \beta_1\beta_2 + 2b\beta_1 + 2b\beta_2 \\
 \gamma_{\phi d1} &= \beta_2 (\beta_1 + \beta_3 + 2\beta_2 - 2b) \\
 \gamma_{\phi d2} &= \beta_1 (\beta_1 - \beta_3 - 2b) - 2\beta_2\beta_3 \\
 \gamma_{\phi d3} &= -(\beta_1^2 + 2\beta_2^2 - \beta_1\beta_3 - \beta_2\beta_3 + \beta_1\beta_2 - 2b\beta_1 - 2b\beta_2)
 \end{aligned}$$

## References

- [1] Panagiotis Tsiotras and Luis Ignacio Reyes Castro. *The Artistic Geometry of Consensus Protocols*, pages 129–153. Springer International Publishing, Cham, 2014.
- [2] Giuseppe Fedele and Luigi D’Alfonso. On the emergent hypocycloidal and epicycloidal formations in a swarm of double integrator agents. *IEEE Control Systems Letters*, 7:613–618, 2023.

- 
- [3] Giuseppe Fedele, Luigi D'Alfonso, and Antonio Bono. Emergent hypotrochoidal and epitrochoidal formations in a swarm of agents. *IEEE Transactions on Automatic Control*, pages 1–8, 2023.
  - [4] Jerome Moses Monsingh and Arpita Sinha. Trochoidal patterns generation using generalized consensus strategy for single-integrator kinematic agents. *European Journal of Control*, 47:84–92, 2019.
  - [5] Jerome Moses Monsingh, Arpita Sinha, and Hoam Chung. Trochoidal patterns generation using generalized consensus strategy for double-integrator dynamic agents. *European Journal of Control*, page 100928, 2023.
  - [6] Aditya Hegde and Anoop Jain. Synchronization and balancing around simple closed polar curves with bounded trajectories and control saturation, 2021. arXiv.2110.07248.
  - [7] Marco Pavone and Emilio Frazzoli. Decentralized policies for geometric pattern formation. In *2007 American Control Conference*, pages 3949–3954, 2007.
  - [8] Aseem Vivek Borkar, Arpita Sinha, Leena Vachhani, and Hemendra Arya. Application of Lissajous curves in trajectory planning of multiple agents. *Autonomous Robots*, 44(2):233–250, January 2020.
  - [9] Manuel Boldrer, Lorenzo Lyons, Luigi Palopoli, Daniele Fontanelli, and Laura Ferranti. Time-Inverted Kuramoto Model Meets Lissajous Curves: Multi-Robot Persistent Monitoring and Target Detection. *IEEE Robotics and Automation Letters*, 8(1):240–247, 2023.
  - [10] P Anjaly and Ashwini Ratnoo. *Analysis of Range-based Agent Motion Patterns*.
  - [11] Anjaly Parayil and Ashwini Ratnoo. Bifurcation-based control law for pattern generation. *IEEE Control Systems Letters*, 3(2):374–379, 2019.
  - [12] James Keller, Dinesh Thakur, Maxim Likhachev, Jean Gallier, and Vijay Kumar. Coordinated Path Planning for Fixed-Wing UAS Conducting Persistent Surveillance Missions. *IEEE Transactions on Automation Science and Engineering*, 14(1):17–24, 2017.