Path Planning for Cooperative Autonomous Soaring gliders

Muhammad Aneeq uz Zaman , Syed Bilal Mehdi , Saad bin Shams

***Abstract*— Soaring refers to the exploitation of free energy available in the environment. One such source of free energy is *thermals* (columns of rising hot air) which are routinely used by birds and glider pilots to increase their flight range. In this paper, we deal with a surveillance problem in which a group of gliders have to visit a set of *interest points*. The gliders can use *thermals* to visit more interest points by increasing their flight range. We present a path planning algorithm which makes sure that the gliders visit as many interest points as possible, while respecting the dynamic constraints of the gliders. We decompose the problem into two parts. The first part deals with planning the best path, for a single glider, while only considering a subset of interest points. The path is planned using Continuous Curvature turns and a graph search based approach. The second part deals with determining the best allocation of interest points for each glider. We also present optimality guarantees for our algorithm.**

I. INTRODUCTION

Recent advances in embedded electronics and advanced control systems has lead to a rapid increase in the usage of Unmanned Aerial Vehicles (UAV)s. Already used by hobbyists UAVs are being proposed for a diverse range of applications, such as precision agriculture [1], public safety applications [2], surveillance [3] and extended coverage [4]. Since several of these applications either require (or benefit from) the UAVs having a high range, methods to increase the range of these UAVs need to be investigated. Furthermore, with a potential explosion in the number of UAVs around the world, increasing flight range can also lead to considerable cost saving for the UAV industry.

Keeping these goals into consideration this paper focuses on one particular method to increase the range of fixed-wing UAVs. Usually referred to as *soaring* this method utilizes prevalent atmospheric phenomenon in order to gain energy during flight. Furthermore, one particular type of soaring that has shown promising results is the use of rising columns of hot air, also known as *thermals*. The use of thermals has been shown to be a viable method of increasing flight range of gliders [5].

This paper considers a particular surveillance problem which requires high range. Specific points on the ground (called *interest points*) have to be visited by a group of gliders. The gliders are constantly losing height in regular flight, but they can use thermals to gain height and hence increase range. The goal of this paper is to plan paths for the gliders so that they visit as many interest points as possible, while using thermals if required. These paths are also required to be valid and feasible. A path is considered *valid* if the glider does not run out of height while traveling on that path. A path is considered *feasible* if the dynamic constraints of the glider are satisfied while executing the path.

This problem is referred to as the multi-agent problem in this paper. To the best of the author's knowledge this particular problem has not been dealt with in literature. This paper proposes a novel approach to solve the problem. This approach entails decomposing the multi-agent problem into multiple single-agent problems.

In a single agent problem a glider is limited to a subset of interest points *allocated* to it. To solve this problem we need to determine a (valid and feasible) path for the glider, which maximizes the number of visited interest points from its allocated interest points. Notice that a glider may not be able to visit all of its allocated interest points due to validity constraints. We first introduce a class of curves which we refer to as legs. A leg provides a feasible path for a glider, going from one waypoint (interest point or thermal) to another. For a path to be feasible its curvature and rate of change of curvature (also called sharpness [6]) should be bounded by predefined constants. The reason for this is discussed in section II. This paper uses continuous curvature (CC) turns [6] to plan feasible paths for the gliders. These legs are then joined together to make paths for each glider, using a modification of the graph-search based approach proposed in paper [7]. The complete path planned for a glider is referred to as a composite path. The proposed algorithm is guaranteed to find the best composite path for a glider.

Now, the multi-agent problem is solved by finding the best allocation for each glider. The allocation which maximizes the number of interest points visited by the gliders collectively, is the best allocation. We also present optimality guarantees for both parts of the algorithm.

A. Related Work

The multi-agent problem is very similar to the Team Orienteering Problem (TOP). In TOP [8] a group of robots have to visit a set of points while minimizing the total distance traveled. Exact approaches to solve the TOP have been proposed in papers [9] and [10]. There has also been work in heuristic methods to solve the TOP [11], [12]. Authors of [7] proposes a graph-search based method. This paper modifies this approach to solve the problem because it is an exact approach and lends itself well to modification.

As for planning feasible paths Dubin's curves have been widely used [13], [14]. Dubin's curves have bounded cur-

Lecturer, National University of Science and Technology, Islamabad, Pakistan, maneeqzaman@gmail.com.

Graduate Student, Dept. of Mechanical Science and Engineering, University of Illinois Urbana-Champaign, mehdi1@illinois.edu.

Lab Engineer, National University of Science and Technology, Islamabad, Pakistan, sbshams@ceme.nust.edu.pk.

vature but unbounded sharpness. Paths with continuous curvature have been proposed by authors of papers [16] and [17], but bounds on sharpness are not considered. Authors of paper [6] propose a suboptimal method for planing paths with bounded curvature and sharpness, and hence we modify this approach to plan feasible paths.

The paper is organized as follows. Section II formulates the problem rigorously. Section III deals with, planning a leg, for a glider, from any given start configuration (position and orientation) to a goal position. Section IV solves the single-agent problem. Section V, describes how to use the method in section IV to solve the multi-agent problem. Section VI presents the simulation results of the proposed solution. Section VII concludes the paper.

II. Problem Statement

This section will present the mathematical formulation for the two (the single-agent and multi-agent) problems separately. First, however, we introduce some details of the problem scenario.

There are a total of n_{ip} interest points each denoted by ip_j where $j \in \{1, .., n_{ip}\}$. These points are located at positions $p_{ip_j} \in \mathbb{R}^2$. There are n_t thermals and each is denoted by t_k where $k \in \{1, .., n_t\}$. The height gained by visiting thermals and the positions of the thermals are assumed to be known a priori. Moreover, the thermals are treated as points and we assume that the gliders gain height instantaneously when they visit a thermal. They are positioned at $p_{t_k} \in \mathbb{R}^2$ and the height any glider can gain by visiting them is denoted by $h_{t_k} \in \mathbb{R}^+$. Similarly we define that the height attained by visiting an interest point is 0, $h_{ip_j} = 0, \forall j$.

We have a total of n_g identical gliders. Each glider i has a pre-specified start position $p_i^0 \in \mathbb{R}^2$ and orientation $\theta_i^0 \in [-\pi, \pi]$. The gliders must reach their respective final positions located at $p_i^f \in \mathbb{R}^2$. The starting height of the gliders is denoted by $h_i^0 \in \mathbb{R}^+$.

Ideally, the height lost per horizontal distance traveled by the glider should be minimized. This corresponds to minimizing the angle of descent $\gamma_{d,i}$ of the glider. As shown in [18] this can be achieved by choosing an appropriate value of angle of attack. For this angle of attack, the corresponding angle of descent is $\gamma_{d_{min}}$. Since the gliders are identical, it is the same for all gliders. We assume that a controller ensures $\gamma_{d,i} = \gamma_{d_{min}}$ for all gliders. Now we may ignore the vertical degree of freedom of the gliders.

A. Single-agent problem

Each glider i is allocated a set of interest points denoted by $\boldsymbol{\xi}_i$, where $\boldsymbol{\xi}_i \subseteq \{ip_1, .., ip_{n_{ip}}\}$. The path glider i takes through $\boldsymbol{\xi}_i$ is called a composite path. The composite path can be composed of multiple legs, where each leg is a path from one waypoint to another.

1) Visitation order: The order in which the glider i visits the waypoints is denoted by $\boldsymbol{\lambda}_i$ such that,

$$\boldsymbol{\lambda}_i = \{\lambda_{i,j}\} : \lambda_{i,j} \in \boldsymbol{\xi}_i \cup \{t_1, .., t_{n_t}\} \cup \{f\}, \quad i \in \{1, .., n_g\}, j \in \{1, .., n(\boldsymbol{\lambda}_i)\},$$

where f represents the final position of the glider and $n()$ is the cardinality operator. $\lambda_{i,j}$ is the jth waypoint in ith glider's composite path. It can only be an interest point allocated to i, a thermal or the final position of i. Notice that a glider does not have to visit all its allocated interest points. The total number of unvisited interest points in $\boldsymbol{\lambda}_i$ is represented by $k(\boldsymbol{\lambda}_i)$,

$$k(\boldsymbol{\lambda}_i) = n(\boldsymbol{\xi}_i) - \sum_{k=1}^{n_{ip}} g(k, \boldsymbol{\lambda}_i) \tag{1}$$

where $g(k, \boldsymbol{\lambda}_i)$ checks if waypoint k has been visited in $\boldsymbol{\lambda}_i$,

$$g(k, \boldsymbol{\lambda}_i) = \begin{cases} 1 & \text{if } k = \lambda_{i,j},\ 1 \le j \le n(\boldsymbol{\lambda}_i) - 1 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

2) Composite path and height profile: Each $\boldsymbol{\lambda}_i$ has a composite path associated with it. The jth leg in the composite path is denoted by $r_{i,j}$. It is parameterized by arc-length of the leg $l_{i,j} \in [0, l_{i,j}^f]$, where $l_{i,j}^f$ is the total arc-length of the leg. The glider dynamics have non-holonomic constraints, meaning $r_{i,j}$ is defined by the orientation of the glider $\theta_{i,j}$,

$$r_{i,j}(l_{i,j}, \boldsymbol{\lambda}_i) = \int_0^{l_{i,j}} \begin{pmatrix} \cos(\theta_{i,j}(u)) \\ \sin(\theta_{i,j}(u)) \end{pmatrix} du + r_{i,j}(0, \boldsymbol{\lambda}_i)\,, \tag{3}$$

$$\kappa_{i,j}(l_{i,j}, \boldsymbol{\lambda}_i) = d\theta_{i,j}(l_{i,j})/dl_{i,j}\,, \tag{4}$$

$$\sigma_{i,j}(l_{i,j}, \boldsymbol{\lambda}_i) = d\kappa_{i,j}(l_{i,j})/dl_{i,j}\,, \tag{5}$$

$$dh_{i,j}(l_{i,j}, \boldsymbol{\lambda}_i)/dl_{i,j} = -\tan(\gamma_{d_{min}})\,, \tag{6}$$

$\kappa_{i,j}$ and $\sigma_{i,j}$ denote the curvature and sharpness of the leg, respectively. $h_{i,j}$ represents the height of the glider, which is decreasing at a constant rate throughout a leg. $\boldsymbol{\lambda}_i$ in the above stated equations may sometimes be omitted for simplicity. The first leg must satisfy start configuration constraints,

$$r_{i,1}(0) = p_i^0, \quad \theta_{i,1}(0) = \theta_i^0, \quad \forall i\,. \tag{7}$$

All the following legs must start at a corresponding waypoint in $\boldsymbol{\lambda}_i$ and maintain continuity of orientation with the last leg,

$$r_{i,j}(0) = p_{\lambda_{i,j-1}}, \quad \theta_{i,j}(0) = \theta_{i,j-1}(l_{i,j}^f)\,. \tag{8}$$

Similarly, each leg j in the composite path should end at either its corresponding waypoint or the final position,

$$r_{i,j}(l_{i,j}^f) = \begin{cases} p_{\lambda_{i,j}} & \text{if } \lambda_{i,j} \neq f \\ p_i^f & \text{if } \lambda_{i,j} = f\,. \end{cases} \tag{9}$$

Each leg should also satisfy the continuity of curvature between consecutive legs,

$$\kappa_{i,j+1}(0) = \kappa_{i,j}(l_{i,j}^f)\,. \tag{10}$$

The glider has a starting height h_i^0 and whenever the glider visits a thermal it gets an increase in its height,

$$h_{i,1}(0) = h_i^0, \quad h_{i,j+1}(0) = h_{i,j}(l_{i,j}^f) + h_{\lambda_{i,j}}\,.$$

3) Feasibility and Validity: The composite paths must satisfy the dynamic constraints on the glider. The gliders have a constraint on the maximum roll angle, which means

that the glider cannot execute tight turns. This corresponds to a maximum curvature constraint on the path. Moreover, there is an upper limit on the roll rate the glider can achieve. This corresponds to an upper limit on the sharpness of the path. Hence, a path must have bounded curvature and sharpness,

$$|\kappa_{i,j}(l_{i,j})| \leq \kappa_{max} : 0 \leq l_{i,j} \leq l^f_{i,j}\,, \forall i, \forall j\,, \quad (11)$$

$$|\sigma_{i,j}(l_{i,j})| \leq \sigma_{max} : 0 \leq l_{i,j} \leq l^f_{i,j}\,, \forall i, \forall j\,. \quad (12)$$

In the paper we assume that κ_{max} and σ_{max} have been predetermined for the gliders. This is called the *feasibility* constraint. Furthermore, the height of the glider must always be positive. This is called the *validity* constraint,

$$h_{i,j}(l_{i,j}) \geq 0 \text{ for } 0 \leq l_{i,j} \leq l^f_{i,j}\,, \forall i, \forall j\,.$$

This can be expressed as $h_{i,j}(l^f_{i,j}) > 0$, since $h_{i,j}$ is constantly decreasing throughout a leg. $h_{i,j}(l^f_{i,j})$ is the height of the glider at the start of the composite path minus the net height lost while traveling,

$$h_{i,j}(l^f_{i,j}) = h^0_i + \sum_{l=1}^{n_t} g(t_l, \boldsymbol{\lambda}_i) h_{t_l} - \tan(\gamma_{d_{min}}) \sum_{k=1}^{j} l^f_{i,k} > 0\,. \quad (13)$$

4) Optimization Problem: The problem now is to find a visitation order which minimizes unvisited allocated interest points. The order should also be complete, meaning that it ends at the final position of the glider. If there are multiple visitation orders which satisfy these criteria, we should find the one with smallest arc-length of the composite path,

$$\underset{\boldsymbol{\lambda}^*_i}{\operatorname{argmin}} \sum_{\forall j} l^f_{i,j} \text{ s.t. } \boldsymbol{\lambda}^*_i = \underset{\boldsymbol{\lambda}_i}{\operatorname{argmin}}\, k(\boldsymbol{\lambda}_i) \quad (14)$$

$$\text{s.t. } \lambda_{i,n(\boldsymbol{\lambda}_i)} = f \text{ and } r_{i,j}(\boldsymbol{\lambda}_i) \text{ satisfies } (11) - (13)\,.$$

The order $\boldsymbol{\lambda}_i$ which satisfies (14), represents the best visitation order for a given waypoint allocation $\boldsymbol{\xi}_i$ to glider i. We call this the optimal visitation order $\boldsymbol{\lambda}^{\boldsymbol{\xi}_i}_i$ for waypoint allocation $\boldsymbol{\xi}_i$. Similarly, the optimal path and the number of unvisited interest points for allocation $\boldsymbol{\xi}_i$ are denoted similarly as $r^{\boldsymbol{\xi}_i}_{i,j}$ and $K(\boldsymbol{\xi}_i)$, respectively.

$$\boldsymbol{\lambda}^{\boldsymbol{\xi}_i}_i = \boldsymbol{\lambda}_i, \quad r^{\boldsymbol{\xi}_i}_{i,j} = r_{i,j}(\boldsymbol{\lambda}_i), \quad K(\boldsymbol{\xi}_i) = k(\boldsymbol{\lambda}_i)\,. \quad (15)$$

B. Multi-agent problem

All $\boldsymbol{\xi}_i$ are mutually exclusive and jointly exhaustive meaning, $\boldsymbol{\xi}_i \cap \boldsymbol{\xi}_j = \emptyset, i \neq j$ and $\bigcup_{i=1}^{n_g} \boldsymbol{\xi}_i = \{ip_1, .., ip_{n_{ip}}\}$, respectively. The set of waypoint allocation for all gliders is denoted by $\boldsymbol{\Xi} = \{\boldsymbol{\xi}_i\}$.

The multi-agent problem is to find the allocations $\boldsymbol{\xi}_i$ with the least number of unvisited interest points. If there are multiple allocations with the same minimum number of unvisited interest points, we should find the one with least accumulated arc-length of composite paths.

$$\underset{\boldsymbol{\Xi}^*}{\operatorname{argmin}} \sum_{i=1}^{n_g} \sum_{\forall j} l^f_{i,j}(\boldsymbol{\lambda}^{\boldsymbol{\xi}^*_i}_i) \text{ s.t. } \boldsymbol{\Xi}^* = \underset{\boldsymbol{\Xi}}{\operatorname{argmin}} \sum_{i=1}^{n_g} K(\boldsymbol{\xi}_i) \quad (16)$$

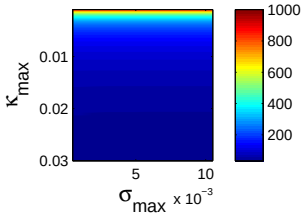

Fig. 1: R_T plotted against κ_{max} and σ_{max}

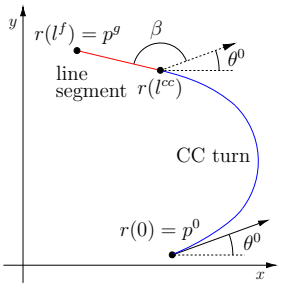

Fig. 2: a leg in the composite path

C. Assumptions

Assumption 1: $\theta_{lim} \triangleq \kappa^2_{max}/\sigma_{max} < \pi$.

Assumption 2: $l_{min} > 2R_T(\kappa_{max}, \sigma_{max})$, where $R_T \in \mathbb{R}^+$ is defined in [6] and l_{min} is the smallest euclidean distance between any two waypoints, start points or final points of any glider.

Assumption 1 is need for the legs to be feasible. If assumption 2 is satisfied, a leg is guaranteed to exist. It might seem, as if assumption 1 can be trivially satisfied by choosing a new maximum curvature $\hat{\kappa}_{max} < \kappa_{max}$. But this might cause us to no longer satisfy assumption 2, as Figure 1 shows that R_T increases as κ_{max} decreases.

III. Planning a leg in the composite path of a glider

This section provides a methodology for planning a leg in a glider's composite path. The purpose of this leg is to provide a path from a start configuration (position and orientation) to a goal position. The methodology is a modification of the approach used in [6]. A leg is composed of two segments: (i) a Continuous Curvature (CC) turn that changes the orientation of the path by an angle β, and (ii) a line segment that completes the leg. The usage of CC turn guarantees that the curvature and sharpness of the leg are bounded, under the assumptions discussed above. An example of the leg obtained by this methodology is shown in Figure 2.

In this section we will focus on one leg denoted by $r(l)$. Subscripts i and j have been omitted for simplicity. The starting position of the leg is denoted by $r(0) = p^0$, starting orientation by $\theta(0) = \theta^0$ and goal position by $r(l^f) = p^g$. Without loss of generality we assume that p^g lies on the left hand side of the start configuration, meaning that the CC turn is anti-clockwise. In case this is not true, a clockwise CC turn is used instead.

A. CC turns

The CC turn starts at $r(0)$ and ends at $r(l^{cc})$ where $l^{cc} < l^f$. The purpose of the CC turn is to change the orientation of the leg until it points to the final position. Therefore, the first step is to calculate the required change in orientation β. The value of β is calculated in Appendix 1 and given by equation (36). This is accomplished by using the fact that the end points of the CC turns (as described in [6]) satisfy

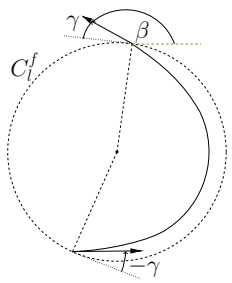

Fig. 3: A CC turn with deflection β

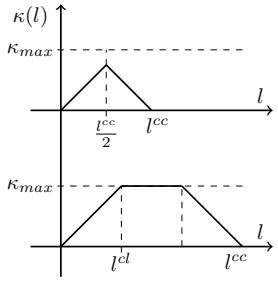

Fig. 4: Curvature profiles of different types of CC turns

two conditions. They exist on the circle C_l^f, and the turn makes an angle γ with C_l^f at the endpoints. The position and radius of C_l^f is dependent on κ_{max} and σ_{max}. A graphical illustration of this condition is provided in figure 3.

There are two types of CC turns depending on the value of β. The arclength of each CC turn, is given as,

$$l^{cc}(\beta) = \begin{cases} 2\sqrt{\beta/\sigma_e} & \text{if } \beta \in [0, \theta_{lim}) \\ \beta/\kappa_{max} + \kappa_{max}/\sigma_{max} & \text{if } \beta \in [\theta_{lim}, 2\pi] \end{cases}. \tag{17}$$

This expression is taken from [6]. The curvature profiles of both CC turns are piece-wise linear as shown in figure 4. For $\beta \in [0, \theta_{lim})$, the curvature profile of the CC turn is,

$$\kappa(l) = \begin{cases} \sigma_e l & \text{for } 0 \leq l \leq l^{cc}/2 \\ \sigma_e(l^{cc} - l) & \text{for } l^{cc}/2 < l \leq l^{cc} \end{cases} \tag{18}$$

where the value of σ_e is given in the paper [6]. σ_e will always exist if assumption 1 is satisfied. If $\beta \in [\theta_{lim}, 2\pi]$, the curvature profile of the CC turn is,

$$\kappa(l) = \begin{cases} \sigma_{max} l & \text{for } 0 \leq l \leq l^{cl} \\ \kappa_{max} & \text{for } l^{cl} < l \leq l^{cc} - l^{cl} \\ \sigma_{max}(l^{cc} - l) & \text{for } l^{cc} - l^{cl} < l \leq l^{cc} \end{cases} \tag{19}$$

where $l^{cl} = \kappa_{max}/\sigma_{max}$. CC turns are guaranteed to satisfy curvature and sharpness constraints (11) and (12), if they exist. This formulation allows for the curvature and sharpness of the path at $l = 0$ and $l = l^{cc}$ to be 0 which automatically satisfies the continuity of curvature constraint in equation (10). The orientation at the end of the CC turn is,

$$\theta(l^{cc}) = \theta(0) + \beta\,. \tag{20}$$

B. Line segment

The next part in the leg is a line segment which starts at $r(l^{cc})$ and ends at $r(l^f)$. The orientation, curvature and sharpness of the line segment is,

$$\theta(l) = \theta(l^{cc}), \quad \kappa(l) = \sigma(l) = 0 \text{ for } l^{cc} < l \leq l^f \tag{21}$$

The total arc length of the leg l^f is derived in appendix 1 and given in equation (34).

Each leg $r(l)$ can be constructed by using the curvature profile $\kappa(l)$ of the leg and its initial conditions.

IV. Single-agent problem

The aim of this section is to solve the single-agent problem. This corresponds to finding a composite path which i) is valid and ii) maximizes the number of (allocated) interest points visited by the glider. This is achieved by choosing the best visitation order, since the composite path is uniquely determined by its corresponding visitation order. The problem of finding the best visitation order is an integer programming problem and we formulate it as a graph search. The graph is dependant on the allocation (of waypoints) $\boldsymbol{\xi}_i$ and denoted by $\Gamma_i(\boldsymbol{\xi}_i)$. Each node in the graph is associated with a visitation order.

This section is organized in the following manner. First, we explain how to construct Γ_i. Next, the algorithm for graph search over Γ_i is presented. This is followed by lemma 1 which states that the graph search finds the node with the best visitation order. Finally, we introduce a measure of 'goodness' of the waypoint allocation $\boldsymbol{\xi}_i$, which is used in section V.

A. Construction of graph $\Gamma_i(\boldsymbol{\xi}_i)$

Let ϕ denote a node in Γ_i. We denote the visitation order associated with ϕ as $\boldsymbol{\lambda}_i(\phi)$. $\boldsymbol{\lambda}_i(\phi)$ has a composite path made up of legs $r_{i,j}(\boldsymbol{\lambda}_i)$ and a height profile $h_{i,j}(\boldsymbol{\lambda}_i)$ associated with it. The legs of the composite path are individually constructed as described in section III with the initial and final conditions being dictated by equations (7)-(9). Its easy to see why the composite path will be uniquely determined by the order $\boldsymbol{\lambda}_i(\phi)$ and the boundary conditions. The root (starting) node of the graph is denoted by ϕ^0. The order associated with this node, $\boldsymbol{\lambda}_i(\phi^0) = \{\}$ is empty and hence there is no composite path. A node is a goal node if last waypoint in its associated visitation order is the final position of the glider. The set of goal nodes is denoted by Φ^f. As in all graphs, goal nodes do not have successors.

The successor to node ϕ is obtained, by appending a leg to the end of composite path $r_{i,j}(\boldsymbol{\lambda}_i(\phi))$. This can be done by adding a thermal, an unvisited allocated interest point or the final position to the end of $\boldsymbol{\lambda}_i(\phi)$. The set of successors of ϕ, $Succ(\phi)$ is defined as,

$$\begin{aligned} \phi' \in Succ(\phi) \text{ only if } \boldsymbol{\lambda}_i(\phi') = \{\lambda_{i,1}(\phi), .., \lambda_{i,n}(\phi), \mu\} \\ \text{s.t. } \mu \in \{\boldsymbol{\xi}_i \cup \{t_1, .., t_{n_t}\} \cup \{f\}\} \setminus \boldsymbol{\lambda}_i(\phi) \end{aligned} \tag{22}$$

where $n = n(\boldsymbol{\lambda}_i(\phi))$, the number of waypoints in $\boldsymbol{\lambda}_i(\phi)$. Now we introduce some values associated with each ϕ which help us determine the best node.

1) Arc length of a node: The arc length of the node, denoted by $S_L(\phi)$, is the total arc length of the composite path associated with $\boldsymbol{\lambda}_i(\phi)$.

$$S_L(\phi) = \sum_{j=1}^{n(\boldsymbol{\lambda}_i(\phi))} l_{i,j}^f\,, \tag{23}$$

where $l_{i,j}^f$ is the arc-length of the jth leg of the glider's composite path. It is defined in equation (34).

2) Cost of a node: Another quantity associated with the a node ϕ is its cost $V_L(\phi)$. The cost of a node determines the order of expansion of the node in the graph search. Nodes with less cost are chosen first. It is defined as,

$$V_L(\phi) = \begin{cases} S_L(\phi) & \text{if } \phi \notin \Phi^f \\ S_L(\phi) + k(\boldsymbol{\lambda}_i(\phi))P_L & \text{if } \phi \in \Phi^f \end{cases}, \tag{24}$$

$$\text{where } P_L = \left(h_i^0 + \sum_{k=1}^{n_t} h_{t_k} + 1 \right) / \tan(\gamma_{d_{min}}) . \tag{25}$$

$k()$ is defined in equation (1). The $k(\boldsymbol{\lambda}_i)P_L$ term, is a penalty term which penalizes the goal nodes with more unvisited allocated interest points. This makes them less likely to be expanded and makes the graph search optimal. For us to find the correct node, P_L has to be an upper bound on the distance the glider can travel. This can be expressed as the upper bound on height of the glider divided by $\tan(\gamma_{d_{min}})$. An upper bound on height of the glider is the starting height of the glider plus the sum of the heights of all thermals $h_i^0 + \sum h_{t_k}$.

3) Validity of a node: A node is considered valid if its corresponding composite path is valid. The validity constraint (13) can be expressed as,

$$S_L(\phi) < \left(h_i^0 + \sum_{l=1}^{n_t} g(t_l, \boldsymbol{\lambda}_i(\phi)) h_{t_l} \right) / \tan(\gamma_{d_{min}}) . \tag{26}$$

B. Uniform cost graph search over Γ_i

Algorithm 1 Uniform Cost Graph Search

Require: $i, p_i^0, h_i^0, p_i^f, \boldsymbol{\xi}_i$
1: $\Omega = \{\phi^0\}, \quad \phi = \phi^0$
2: **while** $\phi \notin \Phi^f$ **do**
3: remove ϕ from Ω with the smallest $V_L(\phi)$
4: **for all** $\phi' \in Succ(\phi)$ **do**
5: calculate $S_L(\phi')$ and $V_L(\phi')$ as given in equations (23) - (24)
6: **if** $S_L(\phi')$ satisfies equation (26) **then**
7: Insert ϕ' into Ω with cost $V_L(\phi')$
8: **end if**
9: **end for**
10: **end while**
11: $\hat{\phi} := \phi$
12: **return** $\hat{\phi}, S_L(\hat{\phi}), k(\boldsymbol{\lambda}_i(\hat{\phi}))$

Algorithm 1 presents the graph search over graph Γ_i. In the algorithm Ω denotes the open set (the set of nodes yet to be expanded), $Vertices(\Gamma_L)$ refers to the set of all possible nodes in Γ_L.

It is a uniform cost search which will return a goal node $\hat{\phi} \in \Phi^f$. For a graph search to be optimal, meaning it finds the node $\hat{\phi}$ s.t. $V_L(\hat{\phi}) \leq V_L(\phi), \forall \phi \in \Phi^f$, the cost of a successor node should be greater than or equal to the cost of the parent node, as shown in the book [19]. For algorithm 1 this is shown easily. Consider a node ϕ and it successor ϕ'. If the $\phi' \notin \Phi^f$, $V(\phi) = S(\phi)$ and $V(\phi') = S(\phi')$. As we know that ϕ' is obtained by adding a leg to the composite path of ϕ, $S(\phi') > S(\phi)$, which leads to, $V(\phi') > V(\phi)$. If the $\phi' \in \Phi^f$, using similar reasoning we arrive at the same conclusion. Hence, $V(\phi') > V(\phi)$. Lemma 1 presents the soundness guarantees for the algorithm, meaning that equations (26) and (14) are satisfied.

Lemma 1: A uniform cost search on the graph Γ_L, as described in Algorithm 1, returns the node $\hat{\phi}$, with $k(\boldsymbol{\lambda}_i(\hat{\phi}))$ number of unvisited interest points. It is guaranteed that (a) $\hat{\phi}$ is a valid goal node, and, (b) $\hat{\phi}$ has the minimum number of unvisited interest points. Moreover, (c) for all nodes in Φ^f with same number of unvisited interest points as $\hat{\phi}$, it has the least arc-length.

Proof: (a) $\hat{\phi}$ is bound to be a goal node, since the *while*-loop at line 4 exits only when a goal node is encountered. Moreover, line 13 in the algorithm makes sure that no node which is invalid is pushed into Ω, hence $\hat{\phi}$ is valid.

(b) We prove by contradiction. Suppose, $\tilde{\phi} \in \Phi^f$, has less unvisited interest points than $\hat{\phi}$, $k(\boldsymbol{\lambda}_i(\tilde{\phi})) < k(\boldsymbol{\lambda}_i(\hat{\phi}))$. Since Algorithm 1 is optimal,

$$V_L(\hat{\phi}) \leq V_L(\tilde{\phi}) ,$$
$$S_L(\hat{\phi}) + k(\boldsymbol{\lambda}_i(\hat{\phi}))P_L \leq S_L(\tilde{\phi}) + k(\boldsymbol{\lambda}_i(\tilde{\phi}))P_L ,$$
$$(k(\boldsymbol{\lambda}_i(\hat{\phi})) - k(\boldsymbol{\lambda}_i(\tilde{\phi})))P_L \leq S_L(\tilde{\phi}) - S_L(\hat{\phi}) ,$$
$$(k(\boldsymbol{\lambda}_i(\hat{\phi})) - k(\boldsymbol{\lambda}_i(\tilde{\phi})))P_L \leq \left(h_i^0 + \sum_{i=1}^{n_t} h_{t_k} \right) / \tan(\gamma_{d_{min}}) .$$

The last step was made possible by the fact that $\left(h_i^0 + \sum_{i=1}^{n_t} h_{t_k}\right) / \tan(\gamma_{d_{min}})$ is an upper bound on $S_L(\phi)$ for any valid node ϕ. Using the definition of P_L and the fact that $k(\boldsymbol{\lambda}_i(\tilde{\phi})) - k(\boldsymbol{\lambda}_i(\hat{\phi})) \geq 1$, we arrive at a contradiction.

(c) Assume another node $\tilde{\phi} \in \Phi^f$, such that $k(\boldsymbol{\lambda}_i(\tilde{\phi})) = k(\boldsymbol{\lambda}_i(\hat{\phi}))$. Since Algorithm 1 is optimal,

$$V_L(\hat{\phi}) \leq V_L(\tilde{\phi}) ,$$
$$S_L(\hat{\phi}) + k(\boldsymbol{\lambda}_i(\hat{\phi}))P_L \leq S_L(\tilde{\phi}) + k(\boldsymbol{\lambda}_i(\tilde{\phi}))P_L ,$$
$$S_L(\hat{\phi}) \leq S_L(\tilde{\phi}) .$$

Hence $\hat{\phi}$ has the minimum arc-length for all goal nodes with number of unvisited interest points equal to $k(\boldsymbol{\lambda}_i(\hat{\phi}))$. ■

C. Arc-length of allocation ξ_i

In the next section we calculate the best allocation of interest points. For that we define the arc-length $S_U()$ of allocation $\boldsymbol{\xi}_i$ as,

$$S_U(\boldsymbol{\xi}_i) = S_L(\hat{\phi}) . \tag{27}$$

V. Multi-agent problem

In this section we solve the multi-agent problem. This corresponds to finding a feasible and valid composite path for each glider such that they collectively visit as many interest points as possible. We decompose this problem into many single-agent problems by allocating a set of interest points to each glider. The method described in section IV, returns the most optimal composite path for that particular allocation of interest points. Using this approach the problem now is

to find the best allocation of interest points for each glider. First we introduce some values associated with each set of allocations $\boldsymbol{\Xi}$, which help us obtain the best interest point allocation.

1) Accumulated arc-length: The accumulated arc-length of a specific $\boldsymbol{\Xi}$ is denoted as $S_U^a(\boldsymbol{\Xi})$ where,

$$S_U^a(\boldsymbol{\Xi}) = \sum_{i=1}^{n_g} S_U(\boldsymbol{\xi}_i)\,, \tag{28}$$

here $S_U(\boldsymbol{\xi}_i)$ is defined in equation (27).

2) Cost of an allocation: Another quantity associated with an allocation is its cost $V_U(\boldsymbol{\Xi})$. It is defined as

$$V_U(\boldsymbol{\Xi}) = S_U^a(\boldsymbol{\Xi}) + P_U \sum_{i=1}^{n_g} K(\boldsymbol{\xi}_i)\,, \tag{29}$$

$$\text{where } P_U = \left(\sum_{i=1}^{n_g} h_i^0 + \sum_{j=1}^{n_t} h_{t_j} + 1\right) / \tan(\gamma_{d_{min}})\,, \tag{30}$$

and $K(\boldsymbol{\xi}_i)$ is the total number of unvisited interest points for allocation $\boldsymbol{\xi}_i$, as expressed in (15). The $P_U \sum_{i=1}^{n_g} K(\boldsymbol{\xi}_i)$ term, is a penalty term which increases the cost of allocations with more unvisited interest points and hence makes them less optimal. To obtain the correct solution, P_U should be an upper bound on the distance the gliders can collectively travel. This can be expressed as an upper bound on collective height of the gliders divided by $\tan(\gamma_{d_{min}})$. An upper bound on height of the gliders is given by $\sum_{i=1}^{n_g} h_i^0 + \sum_{j=1}^{n_t} h_{t_j}$.

The solution to the multi-agent problem is the set of interest point allocations which minimizes the cost V_U and it is denoted by $\hat{\boldsymbol{\Xi}} = \{\hat{\boldsymbol{\xi}}_i\}$. Theorem 1 guarantees that $\hat{\boldsymbol{\Xi}}$ satisfies the optimality criteria (16).

Theorem 1: The set of interest point allocations $\hat{\boldsymbol{\Xi}} = \{\hat{\boldsymbol{\xi}}_i\}$ s.t. $V_U(\hat{\boldsymbol{\Xi}}) \leq V_U(\boldsymbol{\Xi})$, $\forall \boldsymbol{\Xi}$ is guaranteed to (a) have the least number of unvisited interest points and (b) have the least accumulated arc-length of all other $\boldsymbol{\Xi}$ with the same number of unvisited interest points.

Proof: (a) Proof by contradiction. Suppose $\tilde{\boldsymbol{\Xi}} = \{\tilde{\boldsymbol{\xi}}_i\}$, has fewer unvisited interest points $\sum_{i=1}^{n_g} K(\tilde{\boldsymbol{\xi}}_i) < \sum_{i=1}^{n_g} K(\hat{\boldsymbol{\xi}}_i)$. By definition we have,

$$V_U(\hat{\boldsymbol{\Xi}}) \leq V_U(\tilde{\boldsymbol{\Xi}})\,,$$

$$S_U^a(\hat{\boldsymbol{\Xi}}) + P_U \sum_{i=1}^{n_g} K(\hat{\boldsymbol{\xi}}_i) \leq S_U^a(\tilde{\boldsymbol{\Xi}}) + P_U \sum_{i=1}^{n_g} K(\tilde{\boldsymbol{\xi}}_i)\,,$$

$$\left(\sum_{i=1}^{n_g} K(\hat{\boldsymbol{\xi}}_i) - \sum_{i=1}^{n_g} K(\tilde{\boldsymbol{\xi}}_i)\right) P_U \leq S_U^a(\tilde{\boldsymbol{\Xi}}) - S_U^a(\hat{\boldsymbol{\Xi}})\,,$$

$$\left(\sum_{i=1}^{n_g} K(\hat{\boldsymbol{\xi}}_i) - \sum_{i=1}^{n_g} K(\tilde{\boldsymbol{\xi}}_i)\right) P_U \leq \frac{\sum_{i=1}^{n_g} h_i^0 + \sum_{j=1}^{n_t} h_{t_j}}{\tan(\gamma_{d_{min}})}\,.$$

The last step was made possible by the fact that the accumulated arc-length is always upper bounded by the term on the right hand side. Using the definition of P_U and the fact that $\sum_{i=1}^{n_g} K(\hat{\boldsymbol{\xi}}_i) - \sum_{i=1}^{n_g} K(\tilde{\boldsymbol{\xi}}_i) \geq 1$, we get a contradiction.

(b) Assume there is another set of allocation $\tilde{\boldsymbol{\Xi}}$ with the same number of unvisited interest points as $\hat{\boldsymbol{\Xi}}$, $\sum_{i=1}^{n_g} K(\hat{\boldsymbol{\xi}}_i) = \sum_{i=1}^{n_g} K(\tilde{\boldsymbol{\xi}}_i)$. We have,

$$V_U(\hat{\boldsymbol{\Xi}}) \leq V_U(\tilde{\boldsymbol{\Xi}})\,,$$

$$S_U^a(\hat{\boldsymbol{\Xi}}) + P_U \sum_{i=1}^{n_g} K(\hat{\boldsymbol{\xi}}_i) \leq S_U^a(\tilde{\boldsymbol{\Xi}}) + P_U \sum_{i=1}^{n_g} K(\tilde{\boldsymbol{\xi}}_i)\,,$$

$$S_U^a(\hat{\boldsymbol{\Xi}}) \leq S_U^a(\tilde{\boldsymbol{\Xi}})\,.$$

Hence $\hat{\boldsymbol{\Xi}}$ has the minimum accumulated transition cost for equal number of unvisited interest points. ■

VI. Simulation Results

In this section we present an example of cooperative autonomous soaring of 2 gliders with 4 interest points and 4 thermals, meaning $n_g = 2, n_{ip} = 4, n_t = 4$. The maximum curvature $\kappa_{max} = 0.045m^{-1}$ and maximum sharpness $\sigma_{max} = 0.001m^{-2}$. Using these values of κ_{max} and σ_{max},

$$\theta_{lim} = 2.02\ rad,\ R_T = 33.8\ m$$

This satisfies assumption 1. The starting positions, orientations, heights and final positions of the gliders are,

$$p_1^0 = (445, 709)^T m,\ p_2^0 = (646, 754)^T m,$$

$$\theta_1^0 = -1.41 rad,\ \theta_2^0 = 1.13 rad,\ h_1^0 = 600m,\ h_2^0 = 500m,$$

$$p_1^f = (765, 186)^T m, p_2^f = (795, 489)^T m$$

The positions of interest points and thermals are,

$$p_{ip_1} = (97, 950)^T m,\ p_{ip_2} = (823, 34)^T m$$

$$p_{ip_3} = (694, 438)^T m,\ p_{ip_4} = (317, 381)^T m$$

$$p_{t_1} = (743, 706)^T m,\ p_{t_2} = (392, 32)^T m$$

$$p_{t_3} = (655, 277)^T m,\ p_{t_4} = (171, 46)^T m$$

The minimum distance between waypoints $l_{min} = 108.4\ m$ and hence assumption 2 is satisfied. The height a glider can gain by visiting the thermals and interest points is,

$$h_{t_j} = 200m, 1 \leq j \leq 4,\ h_{ip_j} = 0m, 1 \leq j \leq 4.$$

The minimum rate of descent of the gliders is chosen as $\gamma_{d_{min}} = 0.349$ radians.

The set of interest point allocations which minimizes V_U is $\hat{\boldsymbol{\Xi}} = \{\hat{\boldsymbol{\xi}}_i\}$, where the interest points allocated to the first gliders are $\hat{\boldsymbol{\xi}}_1 = \{ip_2, ip_4\}$ and the interest points allocated to the second glider are $\hat{\boldsymbol{\xi}}_2 = \{ip_1, ip_3\}$. The optimal visitation orders for each of these allocations are $\boldsymbol{\lambda}_1^{\hat{\boldsymbol{\xi}}_1} = \{ip_4, t_3, ip_2\}$ and $\boldsymbol{\lambda}_2^{\hat{\boldsymbol{\xi}}_2} = \{t_1, ip_1, ip_3\}$, respectively. The optimal paths associated with $\hat{\boldsymbol{\xi}}_i$, $r_{1,j}^{\hat{\boldsymbol{\xi}}_1}$ and $r_{2,j}^{\hat{\boldsymbol{\xi}}_2}$ are shown in the figure 5.

The curvature, sharpness and height profiles of the optimal paths are shown in figures 6 and 7. The paths are demonstrated to satisfy the endpoint and continuity constraints (7)-(10) and are also feasible and valid as they satisfy (11)-(13).

Figure 8 shows another scenario invloving 2 gliders, 3 interest points and a thermal. Here an interest point in glider

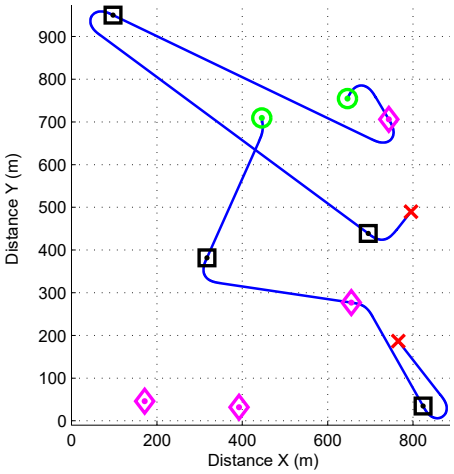

Fig. 5: Cooperative autonomous soaring for 2 gliders, 4 interest points and 4 thermals.

1's composite path is moved such that it comes closer to the composite path of glider 2. The algorithm determines that the more optimal solution where this interest point is in the composite path of glider 2.

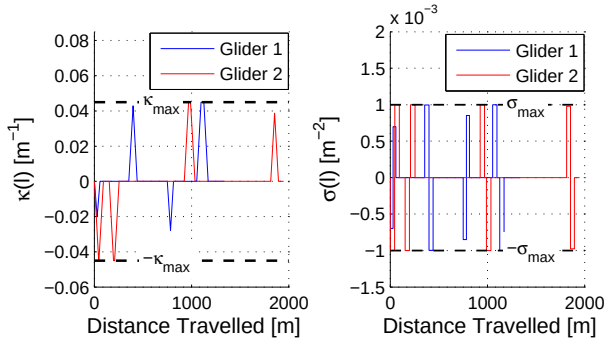

Fig. 6: Curvature and sharpness profiles.

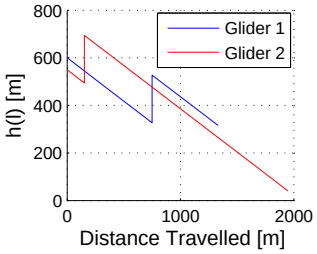

Fig. 7: Height profile.

VII. Conclusion

In this paper we solve the problem of planning paths for multiple gliders which have to visit a set interest points. The gliders can also use thermals to gain height and hence increase their range. First we introduce a special kind of curve, called a leg. Our algorithm uses this leg to generate composite paths for the gliders. The paths planned by the proposed algorithm are shown to be valid and feasible.

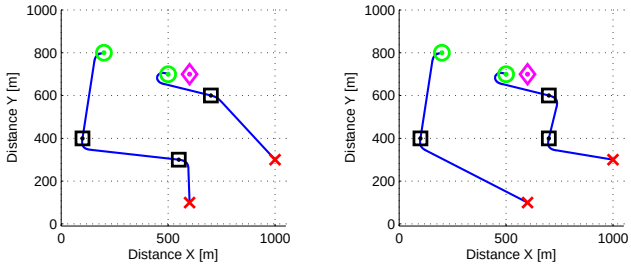

Fig. 8: How moving an interest point changes the optimal solution

Moreover they are guaranteed to cover as many interest points as possible within the particular class of composite paths introduced in the paper. Furthermore, they also have the minimum arc-length of any other path with equal number of visited interest points.

Since, the approach proposed in this paper is exact (meaning it produces the best composite path form a class of composite paths) it is not particularly time efficient. Future work should focus on inexact but faster algorithms. Moreover, since the algorithm produces a composite path from a particular class of composite paths, it is sub-optimal. Hence, efforts should be made to provide bounds on measuring the sub-optimality of the algorithm.

References

[1] Chunhua Zhang and John M Kovacs. The application of small unmanned aerial systems for precision agriculture: a review. *Precision agriculture*, 13(6):693–712, 2012.

[2] Xu Li, Dongning Guo, Huarui Yin, and Guo Wei. Drone-assisted public safety wireless broadband network. In *Wireless Communications and Networking Conference Workshops (WCNCW), 2015 IEEE*, pages 323–328. IEEE, 2015.

[3] Eduard Semsch, Michal Jakob, Dušan Pavlíček, and Michal Pěchouček. Autonomous uav surveillance in complex urban environments. In *Web Intelligence and Intelligent Agent Technologies, 2009. WI-IAT'09. IEEE/WIC/ACM International Joint Conferences on*, volume 2, pages 82–85. IET, 2009.

[4] Peng Cheng, James Keller, and Vijay Kumar. Time-optimal uav trajectory planning for 3d urban structure coverage. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 2750–2757. IEEE, 2008.

[5] Paul B. MacCready. Optimum airspeed selector. *Journal of Soaring*, 1958.

[6] Alexis Scheuer and Th Fraichard. Continuous-curvature path planning for car-like vehicles. In *Intelligent Robots and Systems, 1997. IROS'97., Proceedings of the 1997 IEEE/RSJ International Conference on*, volume 2, pages 997–1003. IEEE, 1997.

[7] Dinesh Thakur, Maxim Likhachev, James Keller, Vijay Kumar, Vladimir Dobrokhodov, Kevin Jones, Jeff Wurz, and Isaac Kaminer. Planning for opportunistic surveillance with multiple robots. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 5750–5757. IEEE, 2013.

[8] Pieter Vansteenwegen, Wouter Souffriau, and Dirk Van Oudheusden. The orienteering problem: A survey. *European Journal of Operational Research*, 209(1):1–10, 2011.

[9] Steven E Butt and David M Ryan. An optimal solution procedure for the multiple tour maximum collection problem using column generation. *Computers & Operations Research*, 26(4):427–441, 1999.

[10] Sylvain Boussier, Dominique Feillet, and Michel Gendreau. An exact algorithm for team orienteering problems. *4or*, 5(3):211–230, 2007.

[11] Moritz Diehl, Hans Georg Bock, Holger Diedam, and P-B Wieber. Fast direct multiple shooting algorithms for optimal robot control. In *Fast motions in biomechanics and robotics*, pages 65–93. Springer, 2006.
[12] Hao Tang and Elise Miller-Hooks. A tabu search heuristic for the team orienteering problem. *Computers & Operations Research*, 32(6):1379–1407, 2005.
[13] Jerome Barraquand and Jean-Claude Latombe. On nonholonomic mobile robots and optimal maneuvering. In *Intelligent Control, 1989. Proceedings., IEEE International Symposium on*, pages 340–347. IEEE, 1989.
[14] Jean-Paul Laumond, Paul E Jacobs, Michel Taix, and Richard M Murray. A motion planner for nonholonomic mobile robots. *Robotics and Automation, IEEE Transactions on*, 10(5):577–593, 1994.
[15] Petr Qvestka and Mark H Overmars. Coordinated motion planning for multiple car-like robots using probabilistic roadrnaps.
[16] Jean-Daniel Boissonnat, André Cerezo, and Juliette Leblond. A note on shortest paths in the plane subject to a constraint on the derivative of the curvature. 1994.
[17] V Kostov and E Degtiariova-Kostova. Irregularity of optimal trajectories in a control problem for a car-like robot. *Rapport de recherche INRIA n 3411*, 1998.
[18] Gerrit JJ Ruijgrok. *Elements of airplane performance*. Delft university press, 1990.
[19] Stuart Russell, Peter Norvig, and Artificial Intelligence. A modern approach. volume 25. Citeseer, 1995.

APPENDIX 1

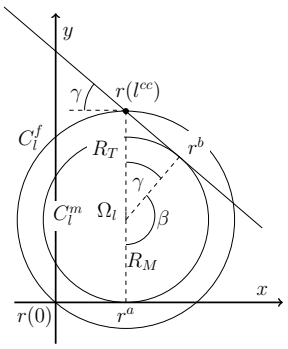

Fig. 9: Lines drawn at $r(0)$ and $r(l^{cc})$ are tangent to the circle C_l^m

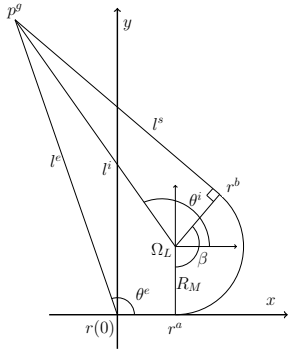

Fig. 10: Lines drawn at $r(0)$ and $r(l^{cc})$, and extended backward are tangent to the circle C_l^m

Without loss of generality we assume that $p^0 = [0,0]^T$ and $\theta^0 = 0$. If this is not true the curve can be obtained by an appropriate amount of translation and rotation of the curve. Lets define θ^e as the angle $\overline{p^g p^0}$ makes with the x-axis. The expression for θ^e is,

$$\theta^e = \arctan\left(\frac{p_x^g}{p_y^g}\right), \tag{31}$$

where p_x^g and p_y^g are the x and y coordinates of p^g, respectively. First we show that the line passing through $r(l^{cc})$ with orientation $\theta(l^{cc})$ is always tangent to a circle C_l^m with origin at Ω_l. This information is then used to obtain the value of β and also the total path length l^f of the leg.

3) Proving tangency to C_l^m: As shown in figure 3 lines passing through $r(0)$ and $r(l^{cc})$ with orientations $\theta(0)$ and $\theta(l^{cc})$ respectively, make an angle γ with the tangent to the circle C_l^f at their respective points. Imagine a circle with radius r and center at point O. If a line intersects this circle at a point, and makes angle γ with the (tangent to the) circle at that point, then this line is tangent to a smaller circle with center at O and radius $r\cos(\gamma)$. Using this fact if we draw lines at points $r(0)$ and $r(l^{cc})$, with orientations $\theta(0)$ and $\theta(l^{cc})$ respectively. These lines will be tangent to another circle, called C_l^m, with center at Ω_L and radius

$$R_M = R_T\cos(\gamma). \tag{32}$$

The lines touch C_L^m at points r^a and r^b respectively. This is depicted in figure 9. Since the lines $\overrightarrow{r^b r(l^{cc})}$ and $\overrightarrow{r(0)r^a}$ make an angle β with each other, lines $\overrightarrow{r^a\Omega_L}$ and $\overrightarrow{\Omega_L r^b}$, will also make an angle β with each other. The straight line $\overrightarrow{r^b r(l^{cc})}$, will intersect any given point on the 2-d plane, for a particular value of β, if its outside of C_L^m. Notice that only the part of this line, starting from $r(l^{cc})$ and going away from r^b, is appended to the CC turn. If assumption 2 is met, p^g will always lie on this part of the line.

4) Calculating arc length of the leg l^f and the angle β: Figure 10 another line segment $\overline{\Omega_L p^g}$, which is added to make calculations easier. The length of the line segment $\Omega_L p^g$ is l^i and the angle it makes with the x-axis is θ^i. l^i can be expressed as,

$$l^i = \sqrt{l^{e2} - 2R_T l^e\sin(\theta^e+\gamma) + R_T^2}. \tag{33}$$

Hence, from figure 10 we see that l^s, which is the length of the segment $\overline{r^b p^g}$, can be expressed as,

$$l^s = \sqrt{l^{i2} - R_M^2}\,,$$

Since, the leg contains the segment $\overline{r(l^{cc})p^g}$, the length of segment $\overline{r^b r(l^{cc})}$ needs to be subtracted from l^s. By observing figure 9, the length of segment $\overline{r^b r(l^{cc})}$ is found to be $R_T\sin(\gamma)$. The length of the leg now becomes,

$$l^f = l^s - R_T\sin(\gamma) + l^{cc}, \tag{34}$$

where l^{cc} is the length of CC turn, given by the equation 17. This length l^f is the arc length of the leg for a glider flying from one configuration to a position.
Moving on to calculating β, from figure 10 the following relation is obtained,

$$l^i\sin(\theta^i) = l^e\sin(\theta^e) - R_M$$

which leads to the expression for θ^i,

$$\theta^i = \begin{cases} \arcsin\left(\frac{l^e\sin(\theta^e)-R_M}{l^i}\right) \\ \qquad \text{if } l^e\cos(\theta^e) - R_T\sin(\gamma) \geq 0 \\ \pi - \arcsin\left(\frac{l^e\sin(\theta^e)-R_M}{l^i}\right) \\ \qquad \text{if } l^e\cos(\theta^e) - R_T\sin(\gamma) < 0\,, \end{cases} \tag{35}$$

the piecewise definition of θ^i is due to the fact that the function arcsin has a range of $[-\pi,\pi]$. Finally, β can be expressed as,

$$\beta\left(p^g, p^0, \theta^0\right) = \theta^i + \arcsin\left(\frac{R_M}{l^i}\right) \tag{36}$$