TECHNICAL RESEARCH REPORT

Robot Formations: Learning Minimum-Length Paths on Uneven Terrain

by Dimitrios (Hristu-Varsakelis) Hristu

CDCSS T.R. 2000-7
(ISR T.R. 2000-25)

CENTER FOR DYNAMICS AND CONTROL OF SMART STRUCTURES

The Center for Dynamics and Control of Smart Structures (CDCSS) is a joint Harvard University, Boston University, University of Maryland center, supported by the Army Research Office under the ODDR&E MURI97 Program Grant No. DAAG55-97-1-0114 (through Harvard University). This document is a technical report in the CDCSS series originating at the University of Maryland.

Web site http://www.isr.umd.edu/CDCSS/cdcss.html

2000 IEEE Mediterranean Conference on
Control and Automation, Rio Patras, Greece

ROBOT FORMATIONS: LEARNING MINIMUM-LENGTH PATHS ON UNEVEN TERRAIN [1]

DIMITRIOS HRISTU(-VARSAKELIS)

University of Maryland at College Park, Institute for Systems Research, College Park, MD 20742, USA, hristu@isr.umd.edu

Abstract. We discuss a prototype problem involving terrain exploration and learning by formations of autonomous vehicles. We investigate an algorithm for coordinating multiple robots whose task is to find the shortest path between a fixed pair of start and target locations, without access to a global map containing those locations. Odometry information alone is not sufficient for minimizing path length if the terrain is uneven or if it includes obstacles. We generalize existing results on a simple control law, also known as "local pursuit", which is appropriate in the context of formations and which requires limited interaction between vehicles. Our algorithm is iterative and converges to a locally optimal path. We include simulations and experiments illustrating the performance of the proposed strategy.

Key Words. Formations, Autonomous Vehicles, Geodesics, Iterative learning.

1 INTRODUCTION

The effort to understand distributed and large-scale systems has brought engineers before a special set of challenges, having to do with the analysis, architecture and of course, control of such enterprises. Recent advances in electronics, computing and wireless communication have made it possible – and indeed practical – to deploy a seemingly endless variety of distributed systems which take advantage of the latest connectivity technologies. Examples of such systems include arrays of satellites, unmanned aerial vehicles (UAVs) and groups of autonomous robots, to name a few.

The importance of these "systems of systems" for a large range of applications (e.g. communications, defense, remote terrain and space exploration), has sparked interest in the problem of effective control and coordination of formations of intelligent machines [11, 10, 5, 3]. Using a *group* (as opposed to a single individual) to accomplish a task, has obvious robustness and redundancy advantages, provided that members of the group can take advantage of each others' presence. In addition, there are tasks (such as the one discussed in this paper) which can be accomplished by a group but not by a single individual. In the following, we investigate one particular type of group behavior sometimes referred to as "local pursuit".

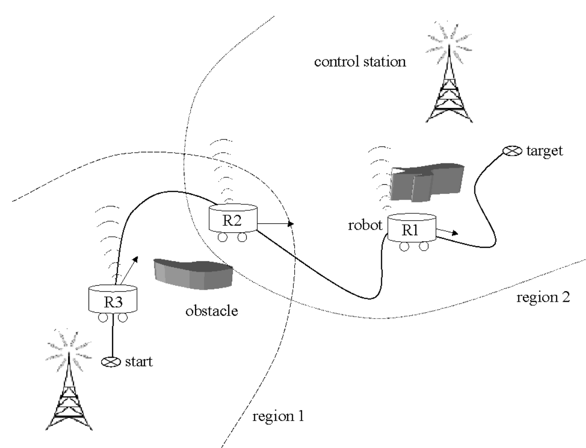

Figure 1: A group of autonomous vehicles

[1] This research was supported in part the National Science Foundation Learning and Intelligent Systems Initiative Grant CMS9720334, and by the Army Research Office under the ODDR&E MURI97 Program Grant No. DAAG55-97-1-0114 to the Center for Dynamics and Control of Smart Structures (through Harvard University).

1.1 Problem description

Consider a group of vehicles moving on terrain which may include geographical formations such as hills, valleys and possibly obstacles (Fig. 1). The group will be required to travel repeatedly to a distant location and back, as it might be the case for successive reconnaissance or sample-collection missions. At least one of the vehicles will have the ability to reach the target (using a combination of prior knowledge, sensor measurements and/or random exploration), possibly taking a circuitous route when doing so. Our goal is to find a strategy by which the vehicles can improve upon that initial path and discover the *shortest path* from their starting location to the target, without access to a map of the area.

The vehicles may have short-range communications and sensors (e.g. odometry, sonar, cameras) with which they can detect each other as well as gather information on their *nearby surroundings*. However, the group has no "global" knowledge of the terrain. In geometric language one might think of the vehicles as moving on a two-dimensional manifold, with each vehicle's sensors covering a local coordinate patch. In that same setting, "shortest" paths are naturally related to the notion of geodesics.

It might be possible to use a "base station" which has knowledge of the terrain, in order to guide the group to the target and back. This approach requires both a detailed map and a significant amount of communication between the centralized controller and every member of the group. This might not be feasible for a variety of reasons, including lack of availability of terrain information, power limitations and stealth constraints.

In Sec. 2.1 we describe an iterative, decentralized approach, which requires interactions between neighboring vehicles only and which converges to a locally optimal path. For the purposes of this work, we model terrain as a smooth surface. In Sec. 3 we present an experiment showing the performance of our algorithm for vehicles whose configuration space is $\mathbb{R}^2$. Section 4 discusses a simulation of the same algorithm on S^2.

2 PURSUIT-BASED OPTIMIZATION

A vehicle designated as the "leader" must first use it sensors, together with any prior information on the target, in order to explore the terrain and arrive at the desired location. The leader's first path to the target will likely be longer than necessary. One would now want to improve on that initial path, minimizing the total length traveled. On flat (e.g. $\mathbb{R}^2$), obstacle-free terrain, the leader could estimate its position and compute a straight path back to the starting location. This would be accomplished by integrating its odometry measurements while searching for the target. However, if the terrain is uneven, determining the shortest path between two points requires a topographic map, which we assume is not available. We now proceed to discuss a strategy which can be used to mitigate the lack of a global map. Our method is based on the work in [2], which we will generalize for terrain with curvature.

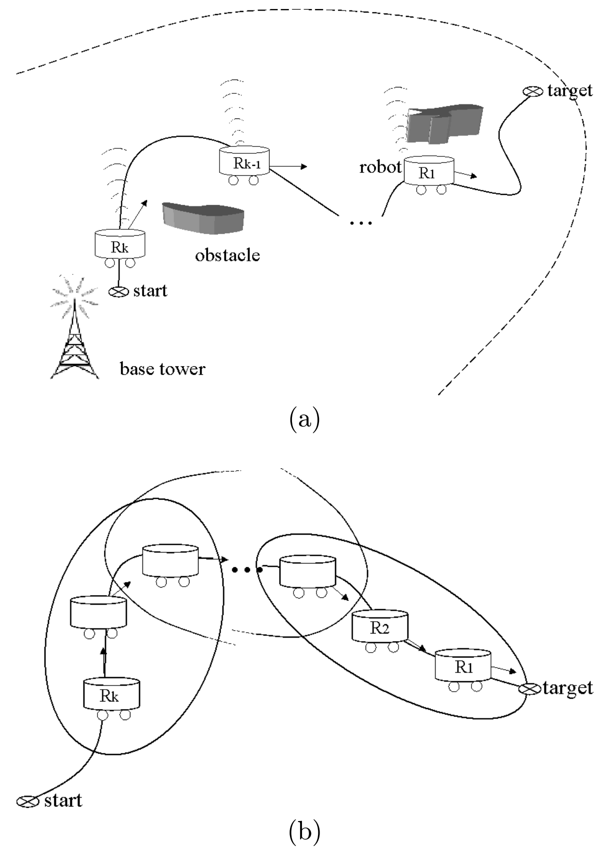

Figure 2: (a) Centralized vs. (b) Decentralized communication

2.1 A local pursuit algorithm

The idea of "local pursuit" involves an ordered sequence of moving vehicles with each vehicle following its predecessor, much like a line of marching ants. From a control-systematic viewpoint, "following" is to be understood as a choice of control inputs which depend on the locations of the pair of vehicles under consideration (i.e. leader and follower).

Initially, all vehicles are at (or near) the starting location. After searching for and finding the target (Sec. 2), a "leader" vehicle reverses course and returns to the rest of the group. Stored odometry information is sufficient for that purpose. Back at the starting location, the leader "recruits" other vehicles to follow it to the target. On level terrain, a vehicle "follows" by pointing its velocity vector in a straight line towards the preceding vehicle [2]. Each vehicle waits Δ units of time before starting its pursuit. We assume that all vehicles move with unit speed. If a vehicle catches up with the one ahead it joins it in its path. For holonomic vehicles which move on $\mathbb{R}^2$, it has been shown [2] that the above strategy gradually "straightens" the iterated paths which connect the starting and target locations, so that the k^{th} robot takes a path which approaches a straight line, for k sufficiently large. Here, we give an extension to more complicated surfaces, replacing the notion of a straight line with that of a geodesic.

2.2 Pursuit on uneven terrain

Ignoring obstacles for the moment, we will model the terrain as a smooth two-dimensional Riemannian manifold M, which we take to be a regular surface, embedded in $\mathbb{R}^3$. We cover M with a set of coordinate neighborhoods parameterized by $u_i = (u_i^1, u_i^2)$. In each neighborhood (corresponding to the coordinate patches surrounding each robot) we choose coordinates $f_i(u_i) = (f_i^1(u_i), f_i^2(u_i), f_i^3(u_i))$. For any two points x, y in some coordinate neighborhood, $\rho(x, y)$ will denote the distance between them. Here, distance is defined as the length of the shortest geodesic connecting x and y, consistent with a choice of inner product on TM. A familiar example is that of of the sphere ($M = S^2$) with the Euclidean metric, in which case the geodesics are simply arcs of great circles on S^2. We will represent the path of the k^{th} robot by a smooth curve $\gamma_k(t) \in M$. We will take all such curves to be parameterized by arclength. Similarly, when referring to time t, we always have in mind that $\dot{t} = 1$. The following result gives sufficient conditions for convergence of the iterated paths generated by the pursuit algorithm:

Theorem 1 *Let M be an m-dimensional Riemannian manifold, parameterized by $u = (u^1, u^2, ..., u^m)$ and let $f(u) = (f^1(u), ..., f^n(u)) \in M \subset \mathbb{R}^n$ be a choice of coordinate functions, $\mathbb{R}^n$ being the "ambient" space ($n > m$). Let p_i, $p_f \in M$ and $\gamma_0(t)$, $t \in [0, L_0]$ a (smooth) curve on M, such that $\gamma_0(0) = p_i, \gamma_0(L_0) = p_f$. Then, the sequence of curves on M defined by*

$$\gamma_{k+1}(t) = p_i \quad if \quad t < (k+1)\Delta \tag{1}$$

$$\dot{\gamma}_{k+1}(t) = \frac{v_{k+1}(t)}{||v_{k+1}(t)||} \quad if \quad t \geq (k+1)\Delta \tag{2}$$

converges to a geodesic if

$$v_k(t) = \frac{d}{ds}\phi_k(t, s)|_{s=0} \tag{3}$$

where $\phi_k(t, s)$ is a geodesic with $\phi_k(t, 0) = \gamma_k(t)$, $\phi\Big(t, \rho\big(\gamma_k(t), \gamma_{k-1}(t)\big)\Big) = \gamma_{k-1}(t)$.

Proof:
Our proof follows that given for $M = R^2$ [2]. Notice that $\phi_k(t, s)$ is the minimum-length geodesic that "connects" the k^{th} and $(k+1)^{th}$ vehicles at time t. In addition, $\phi_k(t, s) = exp(v_k(t), s)$, where $v_k(t) \in T_{\gamma_k}(t)M$ is a tangent vector corresponding to the instantaneous velocity of a pursuing vehicle.

Denote by L_k the total length of the k^{th} path γ_k. We will show that the sequence $\{L_k\}$ is non-increasing. Let $d_i = \rho\big(\gamma_k((k+1)\Delta), \gamma_{k+1}((k+1)\Delta)\big)$ be the initial distance between two consecutive vehicles at the time when the $(k+1)^{th}$ vehicle is leaving the starting point p_i. Departures occur every Δ time units, therefore $0 \leq d_i \leq \Delta$.

Recall that each vehicle moves with unit speed and consider the positions of a leader/follower pair at times t and $t + \epsilon$ (see Fig. 3). Assume for now that at time t,

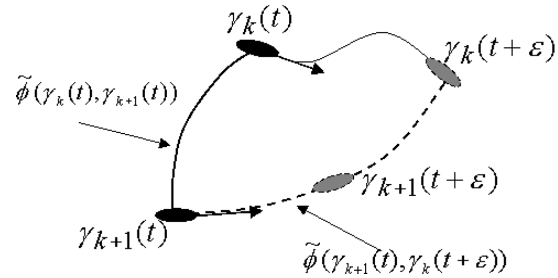

Figure 3: Pursuit strategy on non-flat terrain

the follower knows where the leader will be at the future time $t + \epsilon$. During the interval $[t, t + \epsilon)$, the follower chooses to move along the minimum-length geodesic connecting $\gamma_{k+1}(t)$ to $\gamma_k(t + \epsilon)$ (denoted in Fig. 3 by $\tilde{\phi}(\gamma_{k+1}(t), \gamma_k(t + \epsilon)))$, in an attempt to intercept the leader.

More precisely, we have for all $\epsilon > 0$:

$$\begin{aligned} &\rho\big(\gamma_k(t+\epsilon), \gamma_{k+1}(t+\epsilon)\big) \leq \\ &\quad\leq \rho\big(\gamma_k(t), \gamma_{k+1}(t)\big) + \rho\big(\gamma_k(t), \gamma_k(t+\epsilon)\big) - \epsilon \\ &\quad\leq \rho\big(\gamma_k(t), \gamma_{k+1}(t)\big) + \epsilon - \epsilon \\ &\quad= \rho\big(\gamma_k(t), \gamma_{k+1}(t)\big) \end{aligned} \tag{4}$$

or

$$\frac{\rho\big(\gamma_k(t+\epsilon), \gamma_{k+1}(t+\epsilon)\big) - \rho\big(\gamma_k(t), \gamma_{k+1}(t)\big)}{\epsilon} \leq 0 \tag{5}$$

Now, by letting $\epsilon \to 0$, we have:

$$\frac{d}{dt}\rho\big(\gamma_k(t), \gamma_{k+1}(t)\big) \leq 0 \quad \text{for} \quad t > t_{k+1} \tag{6}$$

When the k^{th} vehicle reaches the target p_f (having traveled L_k units of length), the distance between the k^{th} and $(k+1)^{th}$ vehicles is

$$\rho_f = \rho_i + \int_{t_{k+1}}^{T_k} \rho\big(\gamma_k(t), \gamma_{k+1}(t)\big)dt \leq \rho_i \tag{7}$$

so that the total length traveled by the $k+1$ vehicle is $L_{k+1} \leq L_k$.

Because the departure times are separated by Δ units of time, there will be a finite number of instances where a robot will "catch up" with its leader. We conclude that the sequence L_k must have a limit because it is non-increasing and bounded below. That limit must be a local minimum for the length of the resulting path, otherwise the length of the limiting curve joining p_i, p_f could be further reduced.

In order for equality to hold in Eq. 4, we must have

$$\rho\big(\gamma_{k+1}(t), \gamma_{k+1}(t+\epsilon)\big) = \epsilon \qquad (8)$$

$$\rho\big(\gamma_k(t+\epsilon), \gamma_{k+1}(t+\epsilon)\big) = \rho\big(\gamma_k(t), \gamma_{k+1}(t)\big) \qquad (9)$$

for any $\epsilon > 0$. Equation 8 holds only if γ_{k+1} is a geodesic. Equation 9 tells us that the leader γ_k is also moving along a geodesic. □

2.3 Comments

The direction of pursuit $v_k(t)$ is (locally) optimal, in the sense that if the leader γ_k were to stop, the follower γ_{k+1} would reach the leader by moving on a geodesic.

The implication of taking the limit $\epsilon \to 0$ in Eq. 5, is that the instantaneous velocity of the follower γ_{k+1} approaches the tangent v_k to that geodesic. Thus *when applying Theorem 1, the follower need only know the position of the leader at the current time t and not in any future time*, together with the "best" direction of pursuit $v_k(t)$ (Fig. 4).

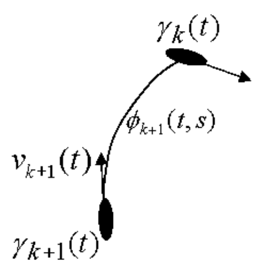

Figure 4: Limit as $\epsilon \to 0$ in Eq. 5

The pursuit strategy outlined in Sec. 2.1, requires making locally optimal decisions that depend on the geometry of a limited region which contains $\gamma_k(t)$ and $\gamma_{k+1}(t)$. Generically, this would be easier than the "global" version of the path optimization problem, because neighboring robots can use sensing (and robot-to-robot communication) to follow one another. It is possible for the pursuit algorithm to converge to the global minimum but this cannot be guaranteed in general. For example, for small values of Δ (vehicle separation), an initial path that winds several times around an obstacle, cannot be expected to "unwind" using local pursuit.

3 EXPERIMENTAL RESULTS

We performed an experiment designed to illustrate the result of Sec. 2, using the robots shown in Fig. 5. Each robot has three wheels, two of which are independently actuated. The wheel configuration makes the robot kinematically equivalent to a unicycle. The robots are outfitted with 16 sonar sensors each as well as odometry sensors and wireless access to the Internet. Their top speed is $2m/s$ and their sensors can be polled at a rate of $30Hz$. In addition, each robot is outfitted with a pair of microphones and speakers. This arrangement allows robots to exchange sound data and get bearing information on one another over short distances.

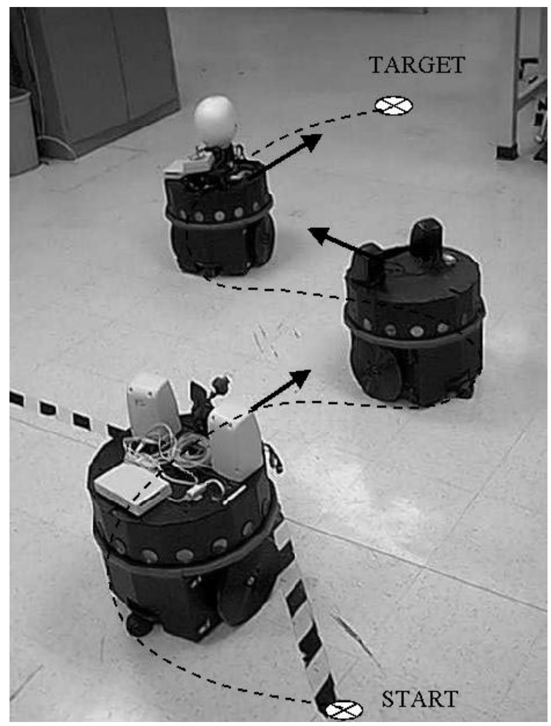

Figure 5: Local pursuit with a trio of robots.

The robots are controlled by means of a Motion Description Language (MDL) [1], which supports interactions between continuous and discrete aspects of a control system. MDL programs are composed by concatenating interrupt-driven "atoms" [6]. Transitions between atoms are triggered by changes in the environment or in the state of the robot. A library of atoms implements simple position and velocity-controlled movements as well as sensing operations involving the sonar, microphones and vision systems.

Our robots were designed for indoor use, therefore the experiment described below was performed on level terrain. We fixed a coordinate frame in the room where the robots were located. Starting at the origin, one of the robots (designated as the leader) was sent out to explore the terrain, recording its odometry data along the way and using its sensors to avoid collisions with obstacles. The leader reached the coordinates $(3.75m, 0.75m)$ which were designated as the target, and returned to the origin by following (backwards) the odometry information it collected on its way to the target. Once back at the origin, the leader turned around and re-traced its original path to the target, this time followed by the two other robots, each separated $0.5m$ from the next. We chose to measure length on the plane using the usual Euclidean metric. This means that the shortest path connecting the origin to the target position was simply a straight line. Each robot followed its leader by moving forward with constant speed, while adjusting its turn rate so as to keep the leader directly ahead. As predicted, each follower robot traveled less distance than its leader, effectively shortening the path between the origin and the target. Once at the target location, the robots followed each other back to the origin, further reducing the total length traveled. We arranged matters so that the robots followed one another back and forth between the origin and the target, in order to circumvent the need for a large number of vehicles, Figure 6 shows the paths traveled by the first (leader) and second robots during seven successive trips between the origin and the target. The curve highlighted with small circles indicates the initial path. As expected, the iterated paths approached a straight line.

It should be mentioned that by choosing R^2 with the Euclidean metric we have effectively ignored the non-holonomic constraint which governs the kinematics of our robots. We were able to do this because following did not require the robots to move sideways. Of course, a more natural choice would have been to regard the robots' configuration space as being $SE(2)$ with an appropriate choice of metric and to look for geodesics in that space [9]. This would involve solving a rather cumbersome two-point boundary value problem on-line, in order to compute the geodesics on $SE(2)$.

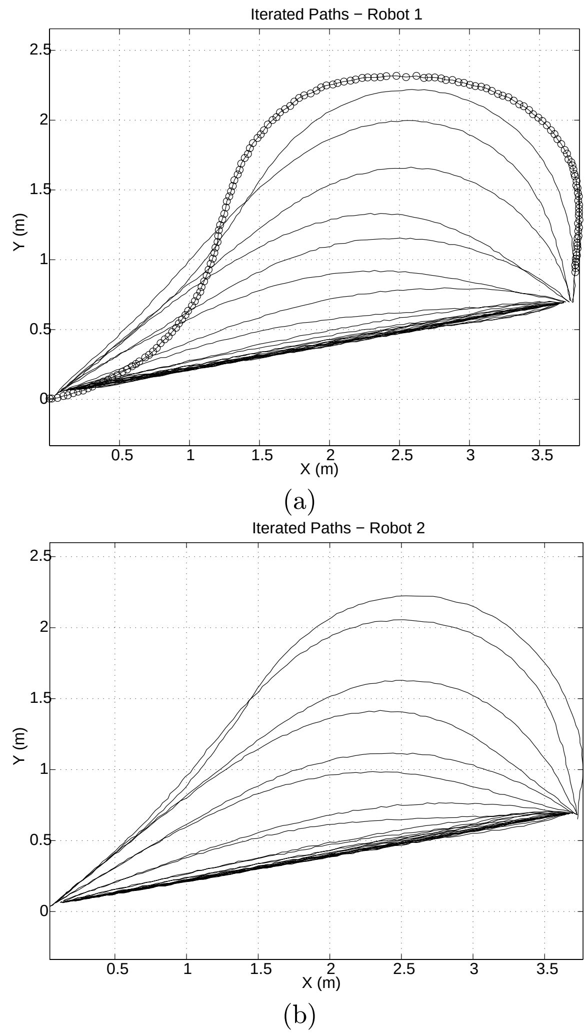

Figure 6: Iterated paths created by "following": (a) first and (b) second robots.

4 SIMULATING PURSUIT ON TERRAIN WITH CURVATURE

To illustrate the pursuit algorithm on a curved surface, we simulated a sequence of (holonomic) vehicles following each other on the unit sphere (with the usual Euclidean metric from $\mathbb{R}^3$). Using spherical coordinates, the start and target locations were given by $(\pi, \frac{\pi}{2})$ and $(0.55, 2.2)$ respectively (numbers refer to radians). The total length of the initial path was 3.8 units of length. Figure 7 shows the evolution of the local pursuit algorithm. The initial path (indicated using small circles) is clearly sub-optimal. Subsequent paths improve the overall length traveled until the path practically becomes a great circle after seven iterations. The separation between vehicles was 0.57 units of length. Here, each vehicle's velocity vector was always tangent to the great circle that connected the vehicle with the one preceding it.

5 CONCLUSIONS

We have explored a cooperative strategy by which a formation of vehicles can progressively "learn" the shortest path between two locations on uneven terrain. Our algorithm is a generalization of "local pursuit", where a sequence of vehicles follow each other from a starting to a target location, much like a group of ants. The paths traversed by the vehicles become progressively shorter, converging to a locally optimal solution without the need for a global map of the terrain. Our algorithm requires an initial (suboptimal) path to be provided or discovered. Each vehicle must be able to compute the

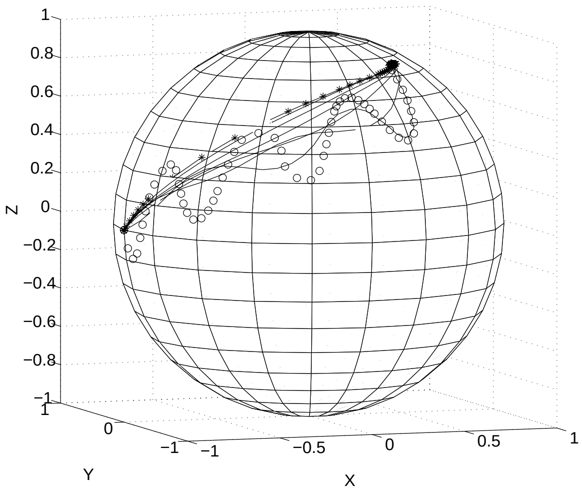

Figure 7: Pursuit on S^2

length-minimizing geodesic passing through its current position and the current position of the preceding vehicle. The notion of geodesic curves and the exponential map allowed us to obtain natural extensions of existing results on planar pursuit problems.

Our result applies to a general class of surfaces, however it requires the ability to consistently choose a length-minimizing curve connecting two nearby vehicles. This may be computationally expensive, depending on the terrain model.

Obstacles were not explicitly considered here but could be included in the analysis as (smooth) deformations of the surface M. The pursuit algorithm would essentially penalize curves that travel over obstacles. Alternatively, one can regard the robots as particles moving among potential fields that emanate from the obstacles [4, 7, 8]. In that case, vehicles are "repelled" as they get close to an obstacle.

Future work will address other interesting aspects of pursuit problems. These include understanding the rate of convergence to the optimal path, the effect of vehicle separation on the convergence rate and the ability of pursuit algorithms to escape local minima. Finally, we intend to investigate the case of non-holonomic vehicles and sub-Riemannian metrics as well as efficient ways to compute geodesics in $SE(2)$ and other spaces.

6 ACKNOWLEDGMENTS

The author would like to acknowledge the help of P. C. Sodre, L. D'Anna and thank professor P. S. Krishnaprasad for his helpful discussions on the subject.

7 REFERENCES

[1] R. W. Brockett. Formal languages for motion description and map making. In *Robotics*, pg. 181–93. American Mathematical Society, 1990.

[2] A. M. Bruckstein. Why the ant trails look so straight and nice. *The Mathematical Intelligencer*, 15(2):59–62, 1993.

[3] J. P. Desai, V. Kumar, and J. P. Ostrowski. Control of changes in formation for a team of mobile robots. In *Proc. 1999 IEEE Int'l Conf. on Robotics and Automation*, vol. 2, pg. 1556–61, May 1999.

[4] Y. K. Hwang and N. Ahuja. Path planning using potential field representation. In *Proc. of the SPIE*, volume 937, pg. 481–8, 1988.

[5] V. Lumelsky and K. Harinarayan. Decentralized motion planning for multiple mobile robots: The cocktail party model. *Autonomous Robots Journal*, (4):121–135, 1997.

[6] V. Manikonda, P. S. Krishnaprasad, and J. Hendler. Languages, behaviors, hybrid architectures and motion control. In J.C. Willems J. Baillieul, editor, *Mathematical Control Theory*, pg. 199–226. Springer, 1998.

[7] H. Noborio, S. Wazumi, and S. Arimoto. An implicit approach for a mobile robot running on a force field without the generation of local minima. In U. Jaaksoo and V. I. Utkin, editors, *Proc. of the 11th Triennial World Congress of the Int'l Federation for Automatic Control*, volume 5, pg. 87–92, 1990.

[8] R. Shahidi. Mobile robot navigation using potential functions. Master's thesis, University of Maryland at College Park, Department of Electrical and Computer Engineering, 1989.

[9] D. Tsakiris. *Motion Control and Planning for Nonholonomic Kinematic Chains*. PhD thesis, University of Maryland at College Park, Department of Electrical and Computer Engineering, 1995.

[10] H. Yamaguchi. A cooperative hunting behavior by mobile robot troops. *Int'l Journal of Robotics Research*, 18(9):931–40, 1999.

[11] H. Yamaguchi and J. W. Burdick. Asymptotic stabilization of mulriple nonholonomic mobile robots forming group formations. In *Proc. 1998 IEEE Int'l Conference on Robotics and Automation*, volume 4, pg. 3573–80, May 1998.