# Rendezvous without Coordinates

Jingjin Yu[†]  Steven M. LaValle[†‡]  Daniel Liberzon[‡]

jyu18@uiuc.edu  lavalle@uiuc.edu  liberzon@uiuc.edu

Department of Computer Science[†]  Coordinated Science Laboratory[‡]
University of Illinois  University of Illinois
Urbana, IL 61801 USA  Urbana, IL 61801 USA

*Abstract*—We study minimalism in sensing and control by considering a multi-agent system in which each agent moves like a Dubins car and has a limited sensor that reports only the presence of another agent within some sector of its windshield. Using a very simple quantized control law with three values, each agent tracks another agent assigned to it by maintaining that agent within this windshield sector. We use Lyapunov analysis to show that by acting autonomously in this way, the agents will achieve rendezvous if the initial assignment graph is connected. A distinguishing feature of our approach is that it does not involve any estimation procedure aimed at reconstructing coordinate information. Our scenario thus appears to be the first example in which an interesting task is performed with extremely coarse sensing and control, and without state estimation. The system was implemented in computer simulation, accessible through the Web [1], of which the results are presented in the paper.

## I. INTRODUCTION

The question of how to perform tasks while requiring a minimal amount of sensing and control information arises in many contexts. Across many subfields of control and robotics, research that focuses on minimalism is of significance for several reasons: 1) It has fundamental importance because it helps to understand the least amount of information or data rate required for a given problem, which in turn offers better insights into a problem's inherent complexity; 2) From a practical standpoint, minimalism in sensing or control immediately leads to less complicated system design, improved robustness, lower production cost, and possibly reduced energy consumption; 3) When physical size of agents is constrained, such as in swarm robotics, sensing and computation become very limited and force the design to go minimal.

In control theory, the problem of controlling a plant using coarse quantized measurements of its state (or output) has received much attention in recent years. Quantized control, an active branch of control theory, focuses particularly on minimal data rate control laws. The motivation for studying such control problems comes from situations in which the rate of information flow between the plant and the controller has to be minimized due to communication bandwidth constraints, shared network resources, security concerns, or other considerations. For some classes of systems, most
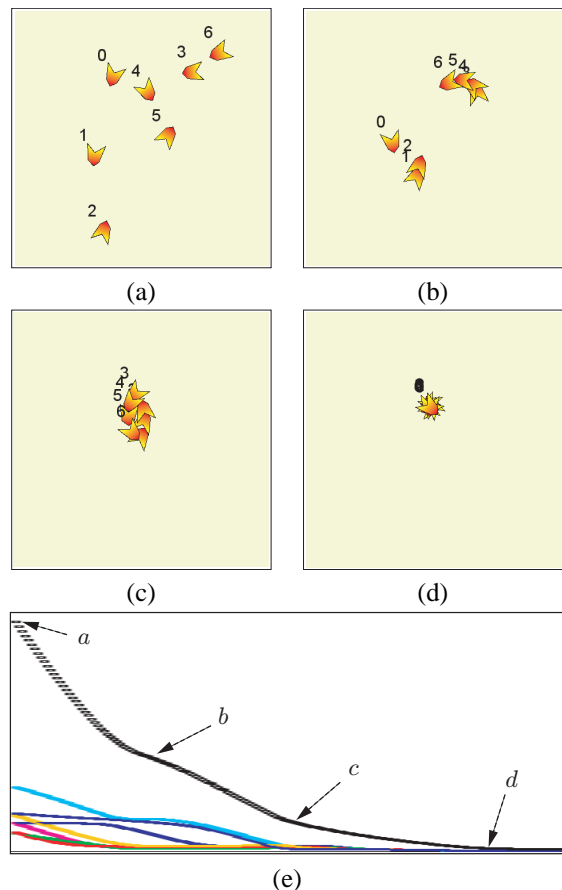


Fig. 1. Screen captures from simulation program. a) - d) Four snapshots of a seven-agent rendezvous, one running after the other. e) Plot of the Lyapunov function of the rendezvous, with *a-d* on the plot corresponding to the four snapshots.

notably linear ones, precise conditions have been obtained on how much information is needed for control; see, e.g., [4], [8], [10], [14], [15], [19], [26], [27], [32], [34]. Minimalism also appears in robotics research that vies for simple abstract sensors. Some earlier efforts are *bug algorithms* for navigation in environments with obstacles [17], [23] and algorithms for manipulating convex polygonal parts using sensorless robots [11], [13]. Recently, *gap navigation trees* for optimal navigation were introduced in [33].

Although quantized control and reduced sensing both emphasize minimal information, they are not minimal in certain

other aspects. Quantized control usually bases the control decisions on estimation of state coordinates, which requires significant computational resources as well as sophisticated analysis tools. Furthermore, to reduce sensing, relatively powerful motion primitives are given to the robots. For example, [33] requires the robot to reliably move toward depth-map discontinuities. Given these developments, it seems that tradeoffs are unavoidable among the various elements of an autonomous system. It is then logical to ask an intriguing question: Is any interesting task even possible with extremely coarse sensing and control, to the point of having only a several-level quantized control but no coordinate data to even enable state estimation?

In this paper we give a "Yes" answer to the above question. We show that, using both simple sensing and quantized control, a group of agents with nonholonomic constraints can rendezvous by simply maintaining target agents within some sector in the "windshield", without ever estimating the current state. Each agent moves as a Dubins car (a restricted unicycle which can move straight or turn left/right with a fixed angular velocity [9], [18]). The agent also has very limited sensors, which cannot measure angles or distances. The sensor output is triggered only when the agent's quantized angle observation changes among one of three possible values. Additionally, agents are not allowed to communicate at all: They are fully autonomous. For this severely restricted system, we find a sufficient condition, using a simple control law, that guarantees rendezvous for any number of randomly placed agents. The results are also verified in simulation (see Figure 1 and Section V).

The rendezvous problem itself has received much attention. A first formulation and algorithmic solution of the multi-agent rendezvous problem is introduced in [1], in which agents have limited range sensing capabilities. Synchronous and asynchronous *Stop-and-go* strategies extending the algorithm in [1] are proposed in [20]. Later, an $n$-dimensional rendezvous problem is approached via proximity graphs in [6]. A research area that is closely related to rendezvous is *cyclic pursuit* or the *n-bug problem*, in which each agent pursues the next one in a cycle. The mathematical study of pursuit curves and pursuit polygons with differential constraints [2], [5], [29] originated this line of research. In the last several years, cyclic pursuit problems with feedback controls were further explored [22], [24], [30], [31]. For history and review on cyclic pursuit, see [24]. Note that the studies mentioned above assume the availability of perfect coordinate information within sensing range, whereas we do not. Other differences include our simple Dubins-based control primitives and our Lyapunov analysis. In particular, we introduce an unusual linear-form (in distance) Lyapunov function that simplifies the analysis. Lyapunov analysis has also been applied to study autonomous group coordination over graphs in [16], but that paper's approach required the agents' awareness of their local environments. Besides [16], distributed consensus has also been focus of [25], [28], [12], [7]. Our study is partly inspired by work on planning for

a differential drive with a limited field-of-view [3], which does not, however, consider minimalism. To the best of our knowledge, there has been no attempt to reduce sensing and control, to the extremely limited combination considered in our paper, to complete interesting tasks, such as rendezvous.

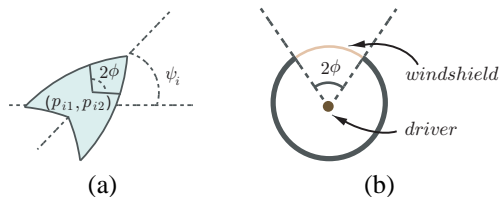## II. PROBLEM FORMULATION

### A. Vehicle model



Fig. 2.  a) An agent's vehicle. b) Agent's windshield, or field-of-view

Consider a set of $n$ *agents*, in which agent $i$ is a point vehicle located at $p_i = (p_{i1}, p_{i2})$ in the plane with orientation $\psi_i$ (Figure 2(a)). Each vehicle moves as a Dubins car:

$$\begin{aligned} \dot{p}_{i1} &= v_i \cos \psi_i \\ \dot{p}_{i2} &= v_i \sin \psi_i \\ \dot{\psi}_i &= u_i, \end{aligned} \tag{1}$$

in which $v_i$ is the constant forward speed, and the only control is $u_i \in \{-\omega_i, 0, \omega_i\}$ for some fixed $\omega_i > 0$. Let $X = SE(2)^n$ denote the *state space*, in which $x \in X$ yields the position and orientation of all agents. The control switches between discrete angular velocities. We sometimes consider agents to be *identical*, which means $v_i = v_j$ and $\omega_i = \omega_j$ for all pairs, $i, j$, of agents.

### B. Sensing model

The vehicle sensor has a limited angular field-of-view, centered at $\psi_i$ with a span $(-\phi, \phi)$ for some given $\phi \in (0, \pi)$; see Figure 2(b). By imagining that you are in the driver's seat, the field-of-view can be considered as a *windshield*. In spite of limited angle, the sensor has unlimited range. Also, an agent cannot occlude another in terms of the sensor view.

For each agent $i$, the sensor does not provide metric information, but instead indicates one of three simple events that may occur with respect to some agent $j$ and the windshield. Let the *observation space* of the sensor be $Y = \{-1, 0, 1\}$, in which an observation $y_i$ for agent $i$ is obtained as

$$y_i = \begin{cases} 1 & \text{agent } j \text{ disappears to the right} \\ 0 & \text{agent } j \text{ appears in the windshield} \\ -1 & \text{agent } j \text{ disappears to the left} \end{cases} \tag{2}$$

Disappearance to the right indicates that agent $j$ was in the windshield and then disappeared across the right windshield boundary. Disappearance to the left is defined similarly. The case of $y_i = 0$ indicates that agent $j$ has appeared in the windshield, from either boundary.

Several implementations of this sensor are possible. Simple image-based detectors could be applied at the two windshield boundaries. To implement it as a simple instantaneous mapping, $h : X \to Y$, the $2\pi$ range of viewing directions could be partitioned into three detection zones, rather than inferring boundary crossing directions, which implies that time history is needed.

In the previous section we mentioned that agents will try to maintain targets in some sector of the windshield; this appears to require the initial condition in which each agent has in the windshield its target. Assume this initial condition for the moment; we later show that this requirement is not necessary.

### C. Expressing a control law

Using (2), a simple sensor-feedback control law $\gamma : Y \to U$ is given by

$$u_i = \omega_i y_i. \tag{3}$$

Using such a simple control law, which involves no state estimation or observation histories, we want to determine the conditions under which agents are guaranteed to rendezvous.

### III. LIVENESS AND CONNECTIVITY

In this section we define a Lyapunov function over a *digraph* (directed graph) $G$, of which each vertex is associated with an agent $i$, which has position $p_i$ and orientation $\psi_i$. We then derive the connectivity condition that $G$ must satisfy for the agents to achieve rendezvous.

### A. A suitable Lyapunov function

For a pair of vertices $i$ and $j$ in a digraph $G$, let $\ell_{i,j}$ denote the distance between the agent positions in $\mathbb{R}_2$. We define a candidate Lyapunov function $V : \mathbb{R}^{2n} \to \mathbb{R}$ as

$$V = \sum_{e_{i,j} \in E(G)} \ell_{i,j}, \tag{4}$$

in which $e_{i,j}$ is the directed edge from vertex $i$ to vertex $j$; $E(G)$ is the edge set of $G$. We call this $V$ a *graph-compatible Lyapunov function* since it is based on some digraph $G$. $V$ is insensitive to edge directions in $G$.

Unlike conventional quadratic Lyapunov functions, this specific $V$ takes a form linear in $\ell_{i,j}$, which means that the corresponding time derivative is not linear in $\ell_{i,j}$ or $p_i$. We justify the choice of this Lyapunov function in the next section. For moving agents, $V$ can be considered a function of time; its time derivative is then

$$\dot{V} = \sum_{e_{i,j} \in E(G)} \dot{\ell}_{i,j}. \tag{5}$$

### B. Liveness and connectivity conditions

To rendezvous, agents must move closer to each other in some way. We say that agent $i$ is *assigned to* agent $j$ if $\dot{p}_i$ is nonzero until $i$ and $j$ rendezvous. One such scenario is when agent $i$ moves at some constant speed and maintains agent $j$ within some sector of its windshield. A special case is when agent $i$ moves directly toward agent $j$. Since $\phi$ may

take values larger than $\pi/2$, assignment alone does not imply a pursuit between agents. We define an *assignment graph $G$* in an obvious way: $G$ initially has $n$ vertices, one for each agent, and there is an edge from $i$ to $j$ if and only if agent $i$ is assigned to agent $j$.

Once agent $i$ catches up with agent $j$, they may combine into a single agent. We call this operation *merging*, which happens within some very small distance $\rho$, in which $\rho$ is the *merging radius*. Merging of $i$ and $j$ is then triggered if

$$e_{i,j} \in E(G) \quad \text{and} \quad \ell_{i,j} \leq \rho. \tag{6}$$

After a merge of agent $i$ reaching agent $j$, vertex $i$ and edge $e_{i,j}$ are deleted from $G$. Any $e_{k,i}$ that existed in $E(G)$ before the merge is replaced by $e_{k,j}$, if such an edge does not already exist. For merging to occur in practice, we may assume that the agents have a short-range proximity detector with the help of which they 'dock' one about the other, but this is not pursued formally.

We say that $G$ is *live* if it has at least one edge. If a graph only has a single vertex, we call it *live* by definition. It is desirable to maintain liveness in $G$ at all times. If $G$ is not live, then it has more than one vertex but no edges; it is then not possible for the system to rednezvous without additional assumptions. Since liveness is not preserved under merging, we need a stronger property from the initial graph.

**Lemma 1** *The assignment graph $G$ is live for all $t \geq 0$, under arbitrary evolution of the agent positions, if and only if it is connected at time $t = 0$.*

PROOF. "If" A connected graph is live and the connectivity of the graph is preserved under merging. (Indeed, connectivity means that there exists a path from any vertex to any other vertex, and merging vertices does not destroy any paths.)

"Only if" If $G$ is not connected, then it has more than one connected component. By merging we can collapse each connected component to a single point. This results in a graph with no edges, which is not live. ∎

By "connected", we mean that the digraph $G$ is at least weakly connected or equivalently, its underlying undirected graph is connected.

We call a candidate Lyapunov function $V$ *rendezvous positive definite* if $V = 0$ while all agents are in the same (unspecified) location, and $V > 0$ otherwise. This leads to:

**Lemma 2** *A graph-compatible Lyapunov function $V$ is rendezvous positive definite if and only if its assignment graph $G$ is connected.*

PROOF. "If" It is clear that $V \geq 0$. Suppose that $G$ is connected and $V = 0$. Since $G$ contains a path connecting all agents, and since the lengths of all edges in this path must be zero, we have rendezvous.

"Only if" If $G$ is not connected, then it has more than one connected component. If each connected component collapses to a single point, $V$ becomes zero, even though

we do not have rendezvous of all agents. ∎

As an alternative to the above direct proof, we could deduce Lemma 2 from Lemma 1: by liveness, the graph will have at least one edge (and thus $V$ will be positive) as long as rendezvous does not occur.
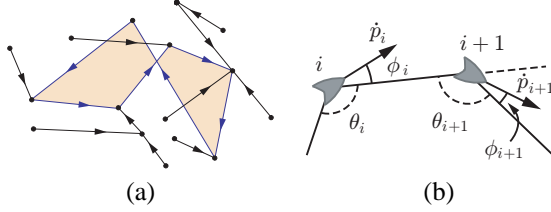


Fig. 3. a) An assignment graph, $G$, with $n$ edges of which the underlying undirected graph is connected and contains a cycle (polygon) that encloses the shaded area. b) Two Dubins-car agents in a cyclic pursuit.

The analysis of Section IV will be organized into two main parts, based on whether $G$ is a tree or is cyclic. By Lemma 1 and 2, for a graph-compatible Lyapunov function to be rendezvous positive definite, the assignment graph $G$ must have at least $n - 1$ edges that span the vertices. The graph $G$ can also have $n$ edges since $G$ has $n$ vertices and every vertex may have one assignment by our limited sensing model and control law. If $G$ has $n - 1$ edges and is connected, it is an *intree* (a directed tree with all paths pointing to its root). The agent at the root of the tree has no target; hence, it does not move. In Section IV, we show that this system must rendezvous, which holds because every agent that is assigned to the root agent will merge into it; the other agents will follow by induction.

If $G$ has $n$ edges, the extra edge induces a cycle; therefore $G$ has a single cycle in which each agent is assigned to the next one. The assignment also guarantees that any agent not on a cycle has a directed path to agents on the cycle (Figure 3(a)). The behavior of the agents on the cycle is not affected by any agent not on the cycle. If the agents on the cycle rendezvous, then they become stationary, and the assignment graph becomes an intree. Thus, we have reduced the rendezvous problem to two subproblems: 1) Rendezvous of agents on a cycle, and 2) rendezvous of agents on an intree.

## IV. GUARANTEED RENDEZVOUS OF AGENTS

### A. The cyclic case

We begin by examining cyclic pursuit in which the pursuit cycle is a simple polygon. Given the Dubins car model (1) and our control law (3), for two consecutive agents $i$ and $i + 1$ on a simple polygon, let $\phi_i$ be the angle between $\dot{p}_i$ and the segment from $p_i$ to $p_{i+1}$ (Figure 3(b)). The angle $\phi_i$ is positive if $\dot{p}_i$ is on the left of the segment from $p_i$ to $p_{i+1}$. Let $\theta_i$ be the internal angle of the simple polygon at vertex $i$. The derivative of the $\ell_{i,i+1}$ is then

$$\dot{\ell}_{i,i+1} = -v_i \cos \phi_i - v_{i+1} \cos(\theta_{i+1} + \phi_{i+1}). \tag{7}$$

After summing up (7) for all $i$ and rearranging, we have

$$\dot{V} = \sum_i - v_i(\cos \phi_i + \cos(\theta_i + \phi_i)). \tag{8}$$

For identical agents with unit speed, (8) becomes

$$\dot{V} = \sum_i - (\cos \phi_i + \cos(\theta_i + \phi_i)). \tag{9}$$

We want to keep $\dot{V}$ negative at all times prior to rendezvous. Note that for $\dot{V}$ in (9):

**Lemma 3** *For any integer $n \geq 2$, the windshield angle $\phi = \pi/n$ permits trajectories for which $\dot{V} = 0$.*

PROOF. The case of $n = 2$ is trivial; hence, assume $n \geq 3$. Since agents may take arbitrary initial formation, they may lie exactly on the vertices of a regular polygon and have the agents on the next vertex as their targets; all agents can also have $\phi_i = \phi = \pi/n$. In this case, since $\theta_i = \frac{n-2}{n}\pi$ for all $i$,

$$\dot{V} = \sum_i - (\cos \frac{\pi}{n} + \cos \frac{n-1}{n}\pi) = 0. \tag{10}$$

∎

For any $\phi \in (\pi/n, \pi]$, trajectories exist for which $\dot{V} \geq 0$ for the same reason. We now give a sufficient condition on $\phi$ for rendezvous for both simple and self-intersecting polygons:

**Theorem 4** *Unit speed cyclic pursuit of $n$ Dubins car agents will rendezvous if the agents maintain their targets in the windshields of span $(-\phi, \phi)$ with:*

$$0 < \phi < \min \left\{ \frac{\pi}{n}, \cos^{-1} \frac{n-1}{n} \right\}. \tag{11}$$

PROOF. We want to show that if $\phi$ satisfies (11), then $\dot{V} < 0$. To facilitate the discussion, define the first and second terms of $\dot{V}$ in (9) as

$$h(\phi_1, \ldots, \phi_n) := \sum_i - \cos \phi_i,$$
$$f(\theta_1, \ldots, \theta_n, \phi_1, \ldots, \phi_n) := \sum_i - \cos(\theta_i + \phi_i). \tag{12}$$

Obviously, $f \leq n$. If for any fixed $n$, there exists some $\delta > 0$, such that $f < n - \delta$, then we can choose some small $\phi$ to make $h$ arbitrarily close to $-n$ to obtain $\dot{V} < 0$. The bound $\phi < \pi/2$ suffices for $n = 2$, which is easy to verify; hence, we work with some $n \geq 3$. We first consider a simple (non-self-intersecting) polygon, which has the property that its internal angles sum up to $(n - 2)\pi$. If we fix $\phi_i$'s in $f$, it is bounded and continuous in $\theta_i$'s; thus, applying the method of Lagrange multipliers with the constraint $g(\theta_1, \ldots, \theta_n) := \sum \theta_i - (n - 2)\pi = 0$ captures all the potential extrema of $f$ for each fixed set of $\phi_i$'s. Examining $\nabla f = \lambda \nabla g$ yields

$$\sin(\theta_i + \phi_i) = \lambda \tag{13}$$

for all $i$ from 1 to $n$. There is an infinite number of solutions to (13), of which most do not correspond to valid simple polygons. Fortunately, we do not need to check many. Lemma 3 suggests that for all $i$, $-\pi/n < (\theta_i + \phi_i) <$

$2\pi + \pi/n$. We partition the $\theta_i$'s, constrained by $g$ and satisfying $0 < \theta_i < 2\pi$, into two sets: $D_{out}$ is the set such that $\theta_i + \phi_i \in (-\frac{\pi}{n}, \frac{\pi}{2}] \cup [\frac{3\pi}{2}, 2\pi + \frac{\pi}{n})$ for some $i$, and $D_{in}$ is the rest, in which all $\theta_i + \phi_i \in (\frac{\pi}{2}, \frac{3\pi}{2})$. We can see immediately that $f \leq n - 1$ on $D_{out}$. Geometrically, $D_{out}$ forms an annulus on the hyperplane $g$, and $D_{in}$ is the inside of that annulus. $D_{in}$ can be empty for $n = 3$. $D_{out} \cup D_{in}$ contains all valid non-self-intersecting polygon angle assignments and maybe some more. By continuity and boundedness of $f$, $f$ is upper bounded on $D_{in}$ by the max among $n - 1$ and extremum of $f$ on $D_{in}$. Since we need $|\phi_i| \leq \phi < \frac{\pi}{n}$, $(n-3)\pi < \sum(\theta_i + \phi_i) < (n-1)\pi$. Solving (13) on $D_{in}$ gives a single extremum in which all $(\theta_i + \phi_i)$'s are equal and less than $\pi$. Hence,

$$f = \sum_i -\cos(\theta_i + \phi_i)$$
$$= -n\cos\frac{(n-2)\pi + \sum\phi_i}{n}. \qquad (14)$$

By (14), this extremum of $f$ only depends on the sum of $\phi_i$'s but not on the individual $\phi_i$'s. Also, fixing $\theta_i$'s, $f$ increases as $\sum\phi_i$ increases; hence, the upper bound of this extremum is given by:

$$f \leq -n\cos\frac{(n-2)\pi + n\phi}{n}. \qquad (15)$$

We obtain for any simple $n$-gon,

$$f \leq \max\left\{-n\cos\frac{(n-2)\pi + n\phi}{n}, n-1\right\}. \qquad (16)$$

Solving $h + f < 0$ with (16) for $\phi$ gives (11). When the polygon is self-intersecting, the internal angles are no longer well defined. However, for our purpose of calculating $\dot{\ell}_{i,j}$, we can always pick the smaller of the two angles at any vertex of the polygon. We call these angles $\theta_i'$ and look at $\sum(\pi - \theta_i')$. If we start from any edge of a polygon, self-intersecting or not, and walk along the edges to get back to the starting edge, all the $(\pi - \theta_i')$'s must add up to at least $2\pi$. This is true since the walker must turn at least a cycle to get back and $\sum(\pi - \theta_i')$ is exactly the sum of angles turned. Thus $\sum\theta_i' \leq (n-2)\pi$. We can then apply the method of Lagrange multipliers similarly. Although we do not have a fixed constraint $g$ in this case, it is only to our advantage: Smaller $\sum\theta_i'$ only loosens the requirement on $\phi$. Thus, (11) still ensures $\dot{V} < 0$.

Having proved that the agents may choose a windshield span size to ensure $\dot{V} < 0$, by the standard Lyapunov theorem on asymptotic stability with respect to a set [21], all agents will rendezvous. The attractive set here is the "diagonal" in $\mathbb{R}^{mn}$, in fact, its $\rho$-neighborhood in which $\rho$ is the merging radius. Note that the introduction of $\rho$ also addresses the issue that our $V$ is not differentiable when some agents are in the same location; however, the result is valid even without this regularization. ∎

Note that for all $n \geq 5$, we have $\pi/n < \cos^{-1}((n-1)/n)$, which implies that the regular n-gon formation maximizes $\dot{V}$ for most $n$ and the condition on $\phi$ is also necessary. By Theorem 4, once $\phi$ is fixed and satisfies (11), the there exists some $\delta > 0$ for which $\dot{V} < -\delta$ for all time. This yields:

**Corollary 5** *System in Theorem 4 achieves rendezvous in finite time.*

By applying the triangle inequality repeatedly, we directly obtain that $\dot{V} \leq 0$ will guarantee a bounding disc of maximum diameter $V$ at all time. Therefore, the same condition will ensure no blowup prior to convergence (i.e., the system is stable in the sense of Lyapunov).

*B. The intree case*

We now look at the case in which the assignment graph is a tree with one agent staying stationary, without the unit speed assumption. Supposing the stationary agent is $j$, at least one agent, say $i$, is assigned to $j$. Thus, $\dot{\ell}_{i,j} = -v_i\cos\phi_i < 0$ whenever $\phi < \pi/2$. Hence, agent $i$ will merge into agent $j$, and any agent after $i$ will eventually follow. We have proved:

**Lemma 6** *Pursuit of Dubins car agents with an intree assignment graph will rendezvous in finite time if the agents maintain their targets in the windshields of span $(-\phi, \phi)$ with $\phi < \pi/2$.*

An overall sufficient condition is then:

**Theorem 7** *Identical Dubins car agents with connected assignment graph will rendezvous in finite time if the agents maintain their targets in the windshields of span $(-\phi, \phi)$ with $\phi$ obeying (11).*

*C. Interesting detour: Direct cyclic pursuit*

If the windshield of an agent is a single point ($\phi = 0$), every agent moves directly toward its target, and we obtain a version of the classic cyclic pursuit, or $n$-bug problem. We call this variety the *direct cyclic pursuit* problem to distinguish it from the Dubins car cyclic pursuit. If we are not sure whether such pursuit is cyclic, we call it *direct pursuit*. Allowing the pursuit formation to be an arbitrary polygon and using the general form of $\theta_i$ defined for self-intersecting polygon, (8) becomes:

$$\dot{V} = \sum_i -v_i(1 + \cos\theta_i), \qquad (17)$$

with $0 < \theta_i \leq \pi$ and $\sum\theta_i \leq (n-2)\pi$. We also allow $v_i$'s to change over time. In (17), $\dot{V}$ is always less than zero. For fixed $n$ and any set of $v_i \geq \epsilon$ for some fixed $\epsilon > 0$, applying the method of Lagrange multipliers shows that there exists some $\delta > 0$ for which $\dot{V} \leq -\delta$ for all time. The steps are similar to that of Theorem 4; the complication introduced by variables $v_i$'s is compensated by the fact that $\phi = 0$. The intree assignment graph case also follows. We obtain the following corollaries:

**Corollary 8** *Agents in a direct cyclic pursuit will rendezvous in finite time.*

**Corollary 9** *Agents in a direct pursuit with an intree connected assignment graph of $n-1$ edges will rendezvous in finite time.*

**Corollary 10** *Agents in a direct pursuit with any connected assignment graph will rendezvous in finite time.*

We can say a little more if $v_i = 1$ holds for all agents:

**Corollary 11** *Agents in a unit speed direct pursuit with connected assignment graph has the property*

$$\dot{V} \leq \max\left\{-1, -n(1+\cos(\frac{n-2}{n}\pi))\right\} := -\delta, \quad (18)$$

*and will rendezvous in time no more than $V_0/\delta$, in which $V_0$ is the value of $V$ at $t=0$.*

### D. Merging radius and angular velocity requirements

Theorem 4 suggests that, if the agents maintain their targets in some $(-\phi, \phi)$ range not violating (11), they will rendezvous. There is a problem, however, since the agents move as Dubins cars which cannot turn arbitrarily fast. As the agents are getting closer, at some point they could form cycles in which they all move along the tangents forever. We resolve this by introducing merging radius $\rho$, with which we remove the arbitrarily fast turning requirement.

**Proposition 12** *In unit-speed Dubins-car pursuit, for any $\rho > 0$, selecting $\omega > 2/\rho$ is sufficient for an agent to track[2] its target or to merge with its target.*
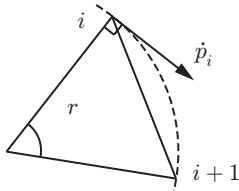
Fig. 4. Dubins car agents that cannot get any closer to their targets.

PROOF. We check the boundary case in which agent $i$ immediately loses sight of agent $i+1$ and can no longer see it by turning around. Since agent $i$ keeps turning, its trajectory is a circle with radius $r = 1/\omega$. On the other hand, agent $i+1$ cannot move faster than agent $i$, so it must move within that same circle to avoid being found (Figure 4). By letting the $\rho > 2r$, we guarantee agents $i$ and $i+1$ will merge. ∎

If all $v_i$ are identical but not equal to one, a similar relationship between $\omega$ and $\rho$ can be obtained.

[2] By tracking a target we mean that once the target is just out of an agent's windshield, the agent can turn fast enough to get its target back into the windshield.

### E. Rendezvous without connectivity

In the set of sufficient conditions stated above that guarantees rendezvous, we have assumed that the assignment graph $G$ is a single component initially. This requires a one time off-line communication of assignment information, or help from an oracle. We now show that the single connected assignment graph assumption is not necessary by allowing agents to choose random targets initially and perform *reassignment*.

By letting the agents choose random targets initially, the assignment graph $G$ can have many components, with each one having a cycle. Look at one such component $G_m$. Once the agents on the cycle of $G_m$ rendezvous and merge, they no longer have any target. We may denote the merged agents as agent $m$. All other agents in $G_m$ will eventually rendezvous with $m$, but there maybe other components that rendezvous at different places in the plane. Thus, agent $m$ should not stay stationary; it needs to determine whether the rendezvous of all components is complete. It can do so by turning around. If agent $m$ sees some agent $j$ in its windshield, it then selects $j$ as a new target and starts pursuing $j$. We call this process *reassignment*. Any agent that has not merged with $m$ can potentially be found by $m$ and get reassigned to $m$; each time, $m$ only needs to turn sufficiently long to detect whether there are further possible reassignments. This is true because as agent $m$ turns around a circle $C$ of radius $\rho/2$, an agent trying to avoid falling into $m$'s windshield must also travel outside $C$ in the same direction, but it cannot do that forever without being found, since it has the same speed as $m$. It can be shown that if $\omega > 3/\rho$, then $m$ needs only to turn two full circles to either find a new target or confirm that it has merged with every other agent. The reassignment algorithm will always keep the graph live until rendezvous. As a last note, agents can also turn in circles to count the total number of other agents to determine the proper windshield size as given by (11). We have shown:

**Proposition 13** *By allowing merging and reassignment for agents that have no targets, unit speed cyclic pursuit of Dubins car agents will achieve rendezvous without initial connectivity requirement.*

### V. IMPLEMENTATION AND SIMULATION RESULTS

In this section we discuss a Java implementation, which we use to demonstrate the theoretical developments. We focus on the two key scenarios of pursuit with cyclic and tree formations. The program was developed adhering to the Java 5 language standard under the Eclipse environment. Results shown here are carried out on a workstation with an Intel Core 2 Quad processor running at 2.4 GHz and 4GB RAM.

Discretization is necessary to implement the continuously evolving system. The sensing needs to be discretized because the Dubins car model allows jumps in angular velocity, which means that there could be a large number of control switches during a short time interval. In our implementation,

we divide the windshield into a left sector and a right sector. The triggering event is simulated by remembering the latest observation of the target agent $j$'s relative position in pursuing agent $i$'s windshield at time step $k$,

$$y_i[k] = \begin{cases} -1 & \text{agent } j \text{ is in the left sector} \\ 0 & \text{agent } j \text{ is out of view} \\ 1 & \text{agent } j \text{ is in the right sector,} \end{cases} \quad (19)$$

and comparing that with the next observation. If the agent is near the center of the windshield, we can give it 1 or $-1$. The quantized control can be encoded as

$$u_i[k] = \begin{cases} -\omega & y_i[k] = 0, y_i[k-1] = -1 \\ 0 & y_i[k] \in \{-1,1\}, y_i[k-1] = 0 \\ \omega & y_i[k] = 0, y_i[k-1] = 1 \\ u_i[k-1] & \text{otherwise.} \end{cases}$$
$$(20)$$

This control law can be expressed as an equivalent four state automaton. The implementation of an agent is therefore also quite simple, which enables the simulation to scale well with respect to the number of agents in the system. Plotting the Lyapunov function for systems with more than 10,000 agents can be handled quickly on our computer with a small memory footprint; smooth animation is possible for 100 agents, with the Lyapunov function and each $\ell_{i,i+1}$ plotted simultaneously in a separate window.
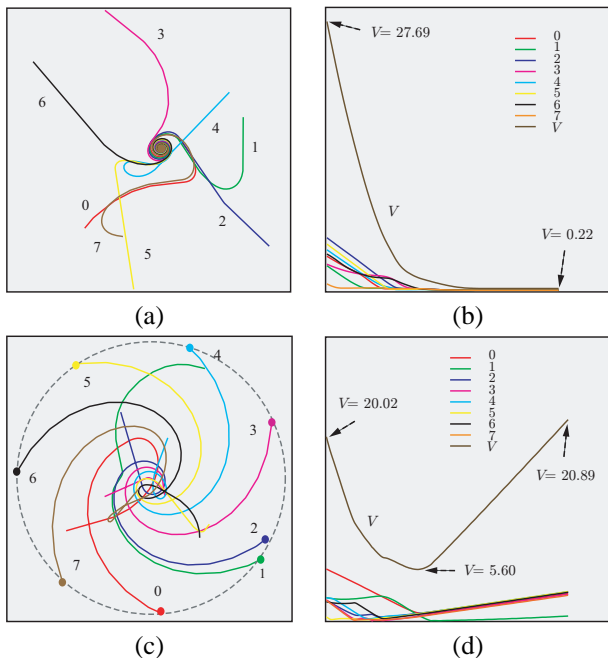


Fig. 5. a) Eight-agent cyclic pursuit, which converges, with $\phi = 0.35$, $\omega = 2.0$, and $\Delta t = 0.1s$. b) Plot of the Lyapunov function for (a), with numbered lines being individual $\ell_{i,i+1}$ terms. c) Diverging case with $\phi$ changed to 0.6 holding other conditions the same. The agents arrange themselves on a cycle quickly as they spin away from each other. Dots are the agents' positions as of the last snapshot. d) Plot of the Lyapunov function for (c).

To verify the cyclic pursuit theorems for the Dubins car, each agent is assigned a unique integer identifier. We initialize the system by positioning the agents randomly and

assigning each agent to the one with a preceding identifier; the first agent is assigned to the last one. The behavior of the cyclic pursuit is then observed by watching the agents' trajectories and a simultaneously drawn plot of the Lyapunov function, under different parameter settings. Since we want to observe the behavior of the cyclic pursuit with all agents, the merging radius $\rho$ is set to near zero to inhibit merging. For every attempted set of $n$ agents, the simulation indicates that $\phi = \pi/n$ is a tight rendezvous bound. As predicted, randomly positioned agents rendezvous when $\phi < \pi/n$, but they will eventually diverge of $\phi$ is only 1% above $\pi/n$. In the latter case, the agents get closer quickly in the beginning because the random arrangement induces a negative $\dot{V}$. Interestingly, the agents then arrange themselves on a circle and eventually form a regular polygon, making $\dot{V}$ positive. We speculate that the continuous system has the same behavior. Additionally, preliminary simulation results show that the system behavior is stable when sensing error is introduced. Figure 5 shows both a converging and a diverging case with $n = 8$. In the rendezvous case, $V$ does not go to zero because of the limit on $\omega$ and discretization.

## VI. CONCLUSION AND FUTURE DIRECTIONS

We have pursued a minimalist approach to a multi-agent rendezvous problem. Using a simple agent model with a ternary output sensor, a three level quantized control, and a Dubins car model, we have shown that a group of such limited agents can still achieve guaranteed rendezvous behavior without coordinates and communication. Furthermore, by exploring the geometric constraints on internal angles of polygons, we have shown that the convergence behavior of the system is precisely predictable based on the parameters of the individual agents. Our simulation indicates that the bounds derived in the theorems may be close to tight; early results from simulation suggest that the simplistic system is robust in the presence of significant sensing errors as well.

Several remaining problems clearly exist. Although we are able to avoid global dynamics arguments, a better understanding of how the system evolves over time will help explain why a regular polygon is the "preferred" formation (the agents seem to form it without much effort). This understanding could also lead to more accurate lower bounds of the time that rendezvous takes for a given arrangement of agents. Another open problem is prescribing the location of rendezvous, which is theoretically appealing and useful for practical purposes.

A couple of immediately ensuing questions are: 1) Is it possible for an even simpler agent model to rendezvous? By simpler we mean that one or more of sensing and control are strictly less powerful, holding the rest of the agent model unchanged. 2) Are there any other tasks achievable with similar simple agents? For example, we see that it is possible for the agents to get into clusters; can they form a regular lattice structure? Can we get them to follow prescribed paths up to homotopy? Or sweeping an area?

Even though we focus on the rendezvous task in this paper, our motivation in this work lies with a more general

goal: Investigating what task classes are possible with minimal amount of information. For a given task, there seems to be an intrinsic relation among the required strengths of the sensors and the controller of an agent. For example, an agent can move to and touch an object to learn its shape; alternatively, it can take a picture and extract the same information. Thus, there must be some equivalence between those two agent models. A firm grasp of this relation will not only help pin down the most basic requirements for a given task, but also offer powerful design guidance for better autonomous systems.

## REFERENCES

[1] H. Ando, Y. Oasa, I. Suzuki, and M. Yamashita. Distributed memoryless point convergence algorithm for mobile robots with limited visibility. *IEEE Trans. Robot. Automat.*, 15(5):818–828, October 1999.

[2] A. Bernhart. Polygons of pursuit. *Scripta Math.*, 24:23–50, 1959.

[3] S. Bhattacharya, R. Murrieta-Cid, and S. Hutchinson. Optimal paths for landmark-based navigation by differential-drive vehicles with field-of-view constraints. *IEEE Transactions on Robotics*, 23(1):47–59, February 2007.

[4] R. W. Brockett and D. Liberzon. Quantized feedback stabilization of linear systems. *IEEE Trans. Autom. Control*, 45(7):1279–1289, 2000.

[5] A. M. Bruckstein, N. Cohen, and A. Efrat. Ants, crickets and frogs in cyclic pursuit. Technical Report CIS Report #9105, Computer Science Department, Israel Institute of Technology, July 1991.

[6] J. Corts, S. Martnez, and F. Bullo. Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions. *IEEE Trans. Autom. Control*, 51(8):1289–1298, August 2006.

[7] A. T. Şamiloğlu, V. Gazi, and A. B. Koku. Asynchronous cyclic pursuit. In *SAB06*, volume 4095 of *Lecture Notes in Artificial Intelligence*, pages 667–678. Springer, 2006.

[8] D. F. Delchamps. Stabilizing a linear system with quantized output record. *IEEE Transactions on Automatic Control*, 35(8):916–926, 1990.

[9] L. E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79:497–516, 1957.

[10] N. Elia and S. K. Mitter. Stabilization of linear systems with limited information. *IEEE Trans. Autom. Control*, 46(9):1384–1400, 2001.

[11] M. A. Erdmann and M. T. Mason. An exploration of sensorless manipulation. *IEEE Transactions on Robotics & Automation*, 4(4):369–379, August 1988.

[12] V. Gazi. Stability of a discrete-time asynchronous swarm with time-dependent communication links. *IEEE Transactions on Systems, Man, and Cybernetics: Part B*, 38(1):267–274, February 2008.

[13] K. Y. Goldberg. Orienting polygonal parts without sensors. *Algorithmica*, 10:201–225, 1993.

[14] J. P. Hespanha, A. Ortega, and L. Vasudevan. Towards the control of linear systems with minimum bit-rate. *in Proc. 15th Int. Symp. Math. Theory Networks and Syst.*, 2002.

[15] H. Ishii and B. A. Francis. Stabilizing a linear system by switching control with dwell time. *IEEE Trans. Autom. Control*, 47(12):1962–1973, December 2002.

[16] A. Jadbabaie, J. Lin, and A. S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Trans. Automat. Control*, 48:988–1001, 2003.

[17] I. Kamon, E. Rivlin, and E. Rimon. Range-sensor based navigation in three dimensions. In *Proceedings IEEE International Conference on Robotics & Automation*, 1999.

[18] S. M. LaValle. *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006. Also available at http://planning.cs.uiuc.edu/.

[19] D. Liberzon. *Switching in Systems and Control*. Birkhäuser, Boston, MA, 2003.

[20] J. Lin, A. S. Morse, and B. D. O. Anderson. The multi-agent rendezvous problem. *Proc. IEEE Conf. Decision and Control*, pages 1508–1513, 2003.

[21] Y. Lin, E. D. Sontag, and Y. Wang. A smooth converse Lyapunov theorem for robust stability. *SIAM J. Control Optim.*, 34:124–160, 1996.

[22] Z. Lin, M. Broucke, and B. Francis. Local control strategies for groups of mobile autonomous agents. *IEEE Trans. Autom. Control*, 49(4):622–629, April 2004.

[23] V. J. Lumelsky and A. A. Stepanov. Path planning strategies for a point mobile automaton moving amidst unknown obstacles of arbitrary shape. *Algorithmica*, 2:403–430, 1987.

[24] J. A. Marshall, M. E. Broucke, and B. A. Francis. Formations of vehicles in cyclic pursuit. *IEEE Trans. Autom. Control*, 49(11):1963–1974, July 2004.

[25] L. Moreau. Stability of multiagent systems with time-dependent communication links. *IEEE Trans. Autom. Control*, 50(2):169–182, February 2005.

[26] G. N. Nair and R. J. Evans. Stabilizability of stochastic linear systems with finite feedback data rates. *SIAM J. Control Optim*, 43:413–436, 2004.

[27] I. R. Petersen and A. V. Savkin. Multi-rate stabilization of multi-variable discrete-time linear systems via a limited capacity communication channel. *in Proc. 40th IEEE Conf. Decision Control*, pages 304–309, 2001.

[28] W. Ren and R. W. Beard. Consensus seeking in multi-agent systems under dynamically changing interaction topologies. *IEEE Trans. Autom. Control*, 50(5):655–661, May 2005.

[29] T. Richardson. Stable polygons of cyclic pursuit. *Annals of Mathematics and Artificial Intelligence*, 31:147–172, 2001.

[30] A. Sinha and D. Ghose. Generalization of linear cyclic pursuit with application to rendezvous of multiple autonomous agents. *IEEE Trans. Autom. Control*, 51(11):1819–1824, November 2006.

[31] S. L. Smith, M. E. Broucke, and B. A. Francis. Curve shortening and the rendezvous problem for mobile autonomous robots. *IEEE Trans. Autom. Control*, 52(6):1154–1159, June 2007.

[32] S. Tatikonda and S. K. Mitter. Control under communication constraints. *IEEE Trans. Autom. Control*, 49(7):1056–1068, July 2004.

[33] B. Tovar, R Murrieta, and S. M. LaValle. Distance-optimal navigation in an unknown environment without sensing distances. *IEEE Transactions on Robotics*, 23(3):506–518, June 2007.

[34] W. S. Wong and R. W. Brockett. Systems with finite communication bandwidth constraints II: Stabilization with limited information feedback. *IEEE Trans. Autom. Control*, 44(5):1049–1053, May 1999.