Controlling Many Differential-Drive Robots with Uniform Control Inputs

Aaron Becker, Cem Onyuksel, Timothy Bretl, and James McLurkin*

Oct 31, 2014

Abstract

This paper derives both open-loop and closed-loop control policies that steer a finite set of differential-drive robots to desired positions in a two-dimensional workspace, when all robots receive the same control inputs but each robot turns at a slightly different rate. In the absence of perturbation, the open-loop policy achieves zero error in finite time. In the presence of perturbation, the closed-loop policy is globally asymptotically stabilizing with state feedback. Both policies were validated with hardware experiments using up to fifteen robots. These experimental results suggest that similar policies might be applied to control micro- and nanoscale robotic systems, which are often subject to similar constraints.

1 Introduction

There are many breakthrough applications for micro- and nanorobotics, across a broad set of fields. Large populations (10^2–10^8) of micromanipulators could allow targeted drug delivery, enable surgeons to operate at the cellular level, let engineers develop more complex MEMS assemblies, and let biologists simultaneously sort all the cells on a Petri dish. These classes of

*A. Becker is with the Department of Cardiovascular Surgery, Boston Children's Hospital and Harvard Medical School, Boston, MA, 02115 USA aaron.becker@childrens.harvard.edu,atbecker@uh.edu; C. Onyuksel is with Western Digital Corporation, Irvine, California 92612 USA cem.onyuksel@gmail.com; J. McLurkin is with the Computer Science Department, Rice University, Houston, TX 77005 USA jmclurkin@rice.edu; T. Bretl is with the Department of Aerospace Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, 61801, USA tbretl@illinois.edu

tasks require *ensemble manipulation*—the use of a large population of small robots to form a coordinated, distributed manipulator.

Ideally, we would design a system that would allow each robot to be controlled individually. However, next-generation micro- and nanorobotic systems are likely to have very minimal on-board processing and communications bandwidth. The lack of significant on-board computation makes autonomous operation infeasible. Sending individual control signals to each robot requires communications bandwidth that scales with population sizes. However, these systems are only useful when their populations are immense, making the bandwidth for individual unit control impractical. A practical solution is to control the entire population with a single, shared, global signal. This implies that all of the robots in the system receive the same control inputs, yet these robots must accomplish different tasks. The solution requires *ensemble control*—the use of a single global control input to direct all of the robots in the population to unique poses.

This paper presents our results on controlling nonholonomic unicycles with uniform inputs:

1. a finite-step open-loop control policy that sends each robot to a.) arbitrary Cartesian positions or b.) range and bearing locations relative to targets in $\mathbb{R}^2$ (Section 3.5).

2. globally asymptotically stabilizing feedback control policy on the position of each robot (Section 4.1).

3. an online calibration procedure to learn model parameters (Section 4.4.3).

4. hardware validations of all control policies (Sections 3.6.2, 4.4.4) with up to $n = 15$ robots that were 0.1m in diameter.

Taken together, this work is an initial step towards practical control for large populations of robots. In the next section we describe the advances in miniature robotics that make control of large robot populations necessary, and the advances in ensemble control that we have applied to this problem.

1.1 Micro and Nano Multi-Robot Systems

Micro- and nanorobots can be produced in extremely large quantities. Once a manufacturing process is developed, the marginal cost of producing one additional robot is small. Microrobots can be fabricated using microelectromechanical system (MEMS) techniques, e.g. scratch-drive micro robots (Donald et al., 2006, 2008, 2013). These robots are 60 by 250 microns in size, and can be mass-produced with many robots tiled on a single silicon wafer. Perhaps the best examples of large populations are robotic *nanocars*—synthetic molecules with integrated axles, rolling wheels, and light-driven molecular motors, that are 1.4×1.7 nm in size. These are routinely produced in tremendous quantities—a batch the size of an aspirin tablet contained $\approx 4 \times 10^{19}$ nanocars (Shirai et al., 2005; Chiang et al., 2011). This dwarfs the total number of birds on the planet earth—some 3×10^{11}(Gaston and Blackburn, 1997). Populations of this size approach a *continuum* of robots. Controlling such a massive population highlights the need for novel approaches for ensemble control, but even populations in the hundreds will not have the processing or bandwidth for individual control.

Instead, this paper focuses on systems with *uniform* control inputs, specifically those that can be modeled as nonholonomic unicycles. Uniform control inputs are common to a variety of biological and artificial robotic systems. Some representative systems are shown in Fig. 1: light-driven nanocars are *uniformly* actuated by a certain wavelength of light, scratch-drive microrobots are *uniformly* actuated by varying the electric potential across a substrate, multi-robot systems are *uniformly* controlled by a broadcast radio signal. Other uniform input examples include the magnetic resonant microrobots of Hsi-Wen et al. (2012), the magnetic helical swimming micro- and nanorobots of both Ghosh and Fischer (2009), Tottori et al. (2012), and Schlüre et al. (2012); the magnetic microparticles of Diller and Floyd et al. (2011; 2012; 2013), the magnetic milli-scale capsules of Vartholomeos et al. (2012), and the tumbling magnetic microrobots of (2013) Biological examples include the electric-field controlled paramecium studied by Hasegawa et al. (2008), galvanotactic the electrokinetic and optically controlled bacteria demonstrated by Steager et al. (2011), the magnetic-field controlled bacteria demonstrated by Lanauze et al. (2013) and magnetic-field steered protozoa demonstrated by Ou et al. (2013). The hardware implementations in this paper use a standard robot platform with the same control inputs simultaneously broadcast wirelessly to all robots (McLurkin et al., 2010).

1.2 Ensemble Control

We limit our work to robots with a differential-drive design, and model their motion as nonholonomic unicycles. If a system of *identical* robots of this type receives the exact same control inputs, their individual positions are not controllable. The path followed by each robot will be a rigid-body trans-

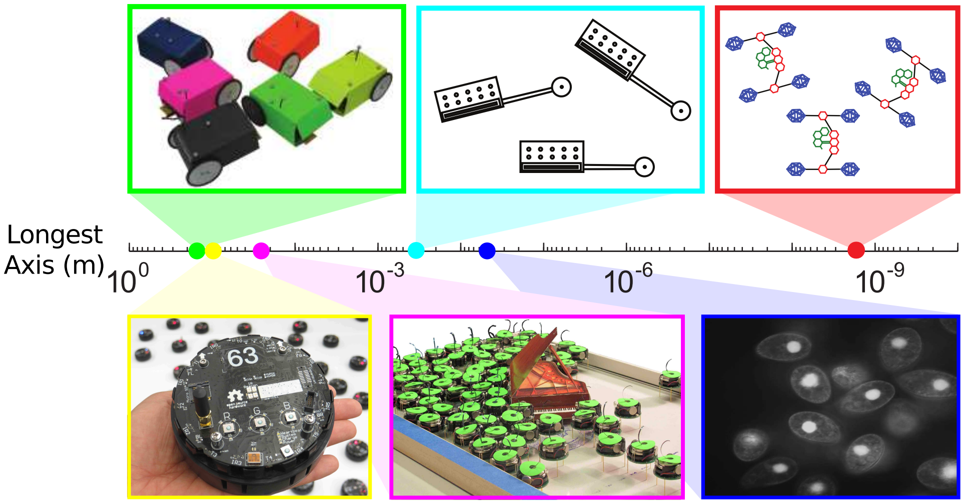


Figure 1: Robotic systems with uniform inputs at decreasing length scales. Top: six differential-drive robots (Becker et al., 2012), three scratch-drive microrobots (Donald et al., 2008), three light-driven nanocars (Chiang et al., 2011). Bottom: swarm of r-one robots (Becker and McLurkin, 2013), photophile kilobot robots (Becker et al., 2013a), and magnetically steered protozoa (Becker et al., 2013b).

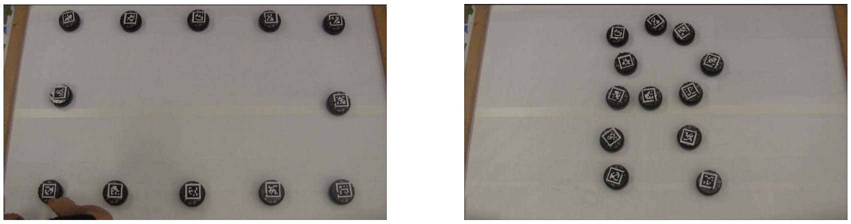
Figure 2: In previous work, we magnetized *T. pyriformis* protozoa (A), and steered them by varying an external magnetic field (B). Because every cell received the same global input signal, rotating the firled casued them all to swim in circles (C)(Becker et al., 2013b). Motion planning for one protozoa is trivial, but joint motion planning for many is complex. add Journal Article

formation of the path followed by every other robot. In practice, however, each robot is slightly different, and this inhomogeneity can be exploited in a systematic way to recover controllability. Inhomogeneities can be found in the systems of Fig. 1 and in other micro- and nanoscale robotic systems. For example, small imperfections in their scratch-drive actuators lead to speed variations between different scratch-drive microrobots. We exploit these differences to develop two controllers: a feed-forward open loop controller suitable if properties of the differences are known a priori, and a feedback controller that uses pose estimates to recover these differences and produce stable control. Though both controllers can exploit random inhomogeneity, we explain that robots with uniform inputs should be *designed* to have large rotational, but small translational process noise.

Previously, we derived an open-loop approximate steering algorithm for one nonholonomic unicycle insensitive to model perturbation, *e.g.* variations in wheel size, that scales both the forward speed and turning rate by an unknown but bounded constant (Becker and Bretl, 2012a). Our solution constructed an infinite, fictitious collection of robots parameterized by the unknown constant, and our control algorithm steered this entire collection to a goal position. Following the terminology introduced by (Brockett and Khaneja, 1999; Khaneja, 2000; Li and Khaneja, 2009; Li, 2011), we called this fictitious collection of unicycles an *ensemble* and called our approach to steering *ensemble control*. The idea was that if the same control inputs steered the entire ensemble from start to goal, then surely they would steer the particular robot of interest from start to goal, regardless of its wheel size.

Our new controllers take advantage of this idea in a slightly different way. Rather than trying to mitigate the effects of bounded model perturbation, we instead try to exaggerate these effects. We will show that if each robot has a different wheel size, we can construct open-loop control policies to drive the robots to desired position or range-and-bearing goal states. Moreover, with state feedback we can derive a globally asymptotically stabilizing control policy that steers the position of all robots (independently) between given start and goal configurations, despite the fact that they all receive the same control inputs.

Ensemble control has recently been applied to a variety of robotics problems, enabling feedforward control of planar robot arms subject to parameter variation (Plooij et al., 2013), simultaneous open-loop control of multiple ball-bearings with unique radii using a plate-ball manipulator (Becker and Bretl, 2012b), and methods for reducing parametric sensitivity applied to feedforward trajectory tracking by robotic vehicles (Ansari and Murphey,



$t = 0$ min $t = 60$ min

Figure 3: n differential-drive robots that receive exactly the same input commands can be steered by our algorithms from any initial position to any desired set of ending positions if robots turn at different rates. Shown are 12 r-one robots moving from a rectangle to an 'R' shape. See Extension 1 for a video of this experiment.

2013). This paper focuses controlling a finite set of robots and provides both an exact open-loop controller, and an asymptotic feedback controller.

1.3 Open-Loop Control

For a system of n robots modeled as nonholonomic unicycles there are $3n$ degrees of freedom: the x, y positions and global headings θ. If the robots have unique turning rates, we provide a finite-time open-loop algorithm to control $2n + 1$ degrees of freedom (DOF). This limitation does not exclude many useful applications, such as navigation, manipulation, or assembly. In this paper we present two useful ways to allocate these $2n + 1$ DOF:

A.) $2n$ unique (x_i, y_i) positions for each robot. The final DOF is the integrated turning commands and modifies the θ_i values, the robot headings in the global reference frame. This behavior is illustrated in Fig. 3.

B.) $2n$ unique (d_i, α_i) range and bearing to targets for each robot. The final DOF is the integrated turning commands and modifies the ψ_i values, the bearings from the target to the robots. We focus on this second behavior, which is illustrated in Figs. 4 and 5.

Heading control is necessary for many robotic tasks, including:

1. receiving an object (fertilizing embryos (Shin et al., 2013), polar-body biopsy, microinjection)
2. redirecting an incoming signal (solar incinerator mirrors)
3. observing an object (collecting, measuring, cameras)
4. emitting an object (ballistics, targeted drug therapy)

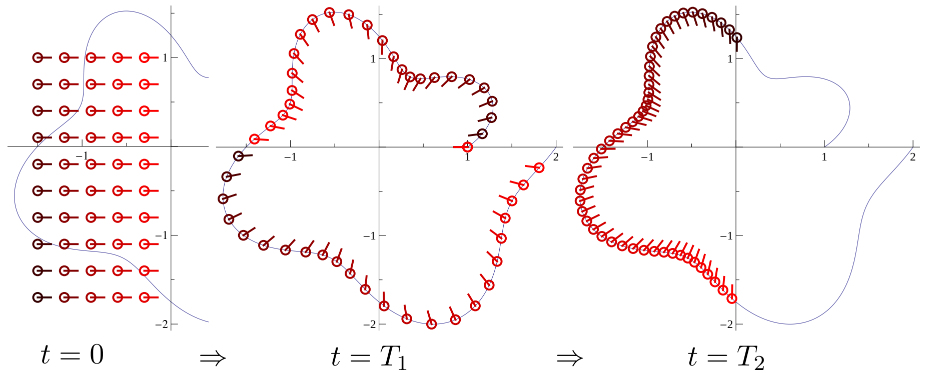
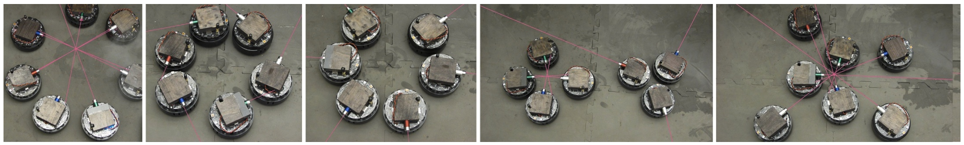


Figure 4: n robots, under the constraint that each robot receives exactly the same control inputs, can be controlled from any initial configuration to the perimeter of a star-shaped set as long as each robot has a unique turning rate. The controller has $2n + 1$ degrees of freedom, which can be used to alter the spacing around the perimeter (middle and right). See Section 3 for details.

5. manipulating objects (pushing, grasping)

On the nanoscale, heading control is no less necessary. Possible applications include using molecular robots as nanoscale transporters, to break chemical bonds, or to build structures by constructing non-covalent bonds. We derive inspiration from the molecular actuators of Minett et al. (2002), the molecular elevators of Badjíc at al. (2004), and the nanocars of Vives et al. (2009).



$d = .3, \alpha = 0$ $d = .2, \alpha = \pi/2$ $d = .25, \alpha = \pi$ $d = \{.1, .05\}, \alpha = \{0, \pi\}$ $d = \{.3, .1\}, \alpha = 0$

Figure 5: Differential-drive robots with top-mounted lasers demonstrating our control technique. The robots receive exactly the same control commands, but each robot has a unique turning rate. By selecting different distances d and bearings $\alpha = \{0, \pi/2, \pi\}$, these robots engage, form a perimeter, or protect the target. These parameters can be mixed and applied to multiple targets, or to form concentric rings.

1.4 Closed-Loop Control

Open-loop control is rarely satisfactory due to model and process noise. Even with exact knowledge of the parameters of each robot, errors due to unmodeled dynamics will integrate over time to ruin the final global configuration. This is particularly troubling for smaller robots, where disturbances from Brownian motion dominate other noise sources (Einstein, 1956). By using a closed-loop feedback policy we can guarantee global asymptotic convergence of the ensemble to any given position. We note that, for single robots, it is possible to build a robust feedback controller that regulates position and orientation (Lucibello and Oriolo, 2001). It is not obvious that the same can be done for an infinite collection of robots. Instead, we focus on regulating the position of every robot in the ensemble, and present simulation results with populations ranging from 120 to 2,000. We also present hardware experiments on populations of up to 15 robots.

1.5 Overview

Our paper is organized as follows. We begin in Section 2 with a description of our problem. In Section 3 we design and evaluate an open-loop controller that assigns $2n+1$ DOF for a n robot system. Section 4 derives a globally asymptotically stable controller. We conclude in Section 5.

Note that preliminary conference versions of Sections 3 and 4 have appeared (Becker et al., 2012; Becker and McLurkin, 2013).

2 Modeling an Ensemble of Unicycle Robots

Consider a collection of n unicycles that each roll without slipping. Following the terminology of (Brockett and Khaneja, 1999; Khaneja, 2000; Li and Khaneja, 2009), we call this collection an *ensemble* and describe the configuration of the ith robot by $\mathbf{q}_i = [x_i, y_i, \theta_i]^\top$ and its configuration space by $\mathcal{Q} = \mathbb{R}^2 \times \mathbb{S}^1$. The global control inputs are the forward speed $u \in \mathbb{R}$ and turning rate $\omega \in \mathbb{R}$. We assume that each robot has a nonzero parameter v_i that scales the linear velocity and a unique nonzero parameter ϵ_i that scales the turning rate ($|\epsilon_i| \neq |\epsilon_j| \; \forall \; i, j$). These v_i, ϵ_i values may arise from stochastic processes during manufacturing (Donald et al., 2006), or as design decisions (Peyer et al., 2013). The kinematics of the unicycle are given by

$$\dot{\mathbf{q}}_i(t) = v_i u(t) \begin{bmatrix} \cos\theta_i \\ \sin\theta_i \\ 0 \end{bmatrix} + \epsilon_i \omega(t) \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \tag{1}$$

If v_i is zero the robot cannot move. Similarly, $\epsilon_i = 0$ prevents the robot from turning. On a collection of differential-drive robots, these parameters can be mapped to unique wheel sizes and $\epsilon_i = v_i$.

2.1 Conversion to Discrete-Time

We model our robotic system with a discrete-time model. We can simplify (1) by splitting each ΔT time step into two stages with piecewise constant inputs. During the first stage of round k we command the robots to turn in place ϕ, and during the second stage command the linear movement $u(k)$.

$$k = \left\lfloor \frac{t}{\Delta T} \right\rfloor$$

$$\Big[u(t), \omega(t)\Big] = \begin{cases} \left[0, \quad \frac{2}{\Delta T}\phi\right] & t - k\Delta T < \frac{\Delta T}{2} \\ \left[\frac{2}{\Delta T}u(k), 0\right] & \text{else} \end{cases} \tag{2}$$

Because the robots are either turning in place or moving in a straight line, we can precompute the heading angles and write the kinematics in the following simple form

$$\begin{bmatrix} x_i(k+1) \\ y_i(k+1) \end{bmatrix} = \begin{bmatrix} x_i(k) \\ y_i(k) \end{bmatrix} + \begin{bmatrix} v_i \cos\left(\theta_i(0) + \epsilon_i k \phi\right) \\ v_i \sin\left(\theta_i(0) + \epsilon_i k \phi\right) \end{bmatrix} u(k), \tag{3}$$

for $i = 1, 2, \ldots, n$ and $k \in \mathbb{Z}$. Eq. (3) is a discrete-time linear time-varying system. As $\Delta T \to 0$, the discrete-time ensemble (3) approaches the continuous-time model (1). Note that, alternatively, u could be constant and $\phi(k)$ the independent variable, but the resulting system would be nonlinear. This linear formation (3) converts the motion planning problem for n robots with uniform inputs into a matrix inversion, as shown in the following section.

3 Open-Loop Ensemble Control

We present two variants of open-loop control: position control and range-and-bearing control. Figure 3 shows the output of position control. Note that the robots reach desired final positions, but end with seemingly random headings. This reflects the $2n + 1$ controllable degrees of freedom: the final positions are controllable, but the final headings are linked. Figure 4 shows range-and-bearing control. Note that the robots reach desired range and bearings, but their ending positions are distributed along the perimeter. This also reflects the $2n + 1$ controllable degrees of freedom: n arbitrary range and bearings are controllable, but the final spacings are linked.

3.1 Position Control

To prove (3) is controllable by (2), we will show the system is **uniformly k-step controllable**. This means that the reachable set after k rounds is the entire state-space (Levine, 1996, chap 25.3). We define the state as the robot position $\mathbf{p}_i = [x_i, y_i]^\intercal$, and write (3) in standard notation as

$$\mathbf{p}_i(k+1) = A_i(k)\mathbf{p}_i(k) + B_i(k)u(k). \tag{4}$$

Here $A_i(k)$ is the identity matrix for all i, k and $u(k)$ the commanded linear movement. The matrix $B_i(k)$ encodes all heading information, $\theta_i(k)$, and has the form

$$\begin{aligned}
B_i(0) &= v_i \begin{bmatrix} \cos(\theta_i(0)) \\ \sin(\theta_i(0)) \end{bmatrix} \\
B_i(1) &= v_i \begin{bmatrix} \cos(\theta_i(0) + \epsilon_i \phi) \\ \sin(\theta_i(0) + \epsilon_i \phi) \end{bmatrix} \\
&\vdots \\
B_i(k) &= v_i \begin{bmatrix} \cos\left(\theta_i(0) + \epsilon_i k \phi\right) \\ \sin\left(\theta_i(0) + \epsilon_i k \phi\right) \end{bmatrix}, \\
\mathbf{B}(k) &= \begin{bmatrix} B_1(k) \\ B_2(k) \\ \vdots \\ B_n(k) \end{bmatrix}.
\end{aligned}$$

We then define the controllability matrix $\mathcal{C}_k$ as

$$\mathcal{C}_k = \left[\mathbf{B}(0), \mathbf{B}(1), \ldots, \mathbf{B}(k-1)\right].$$

The finite ensemble with n robots has $3n$ degrees of freedom, but we can control only $2n+1$ of these. We choose to control the x, y positions and the net turning command $k\phi$. To control each robot's x, y position requires $\mathcal{C}_k$ to be rank $2n$. If the ϵ_i values that scale the turning speed are unique, the functions $\cos(\epsilon_1\phi)$,$\sin(\epsilon_1\phi)$, $\ldots$, $\cos(\epsilon_n\phi)$,$\sin(\epsilon_n\phi)$ are orthogonal on any closed interval of length 2π. This means there always exists a sequence of ϕ values that make $\mathcal{C}_k$ full rank. The parameter ϕ controls the sampling frequency, and must be twice the Nyquist frequency(Shannon, 1949), or

$$\phi \le \frac{\pi}{\max\limits_{i \in [1,n]} \epsilon_i}. \tag{5}$$

Sampling theory also gives a bound on k, the number of samples needed (Proakis and Manolakis, 1996, Chapter 5). In order to differentiate two frequencies ϵ_1 and ϵ_2, we need $k \geq \frac{2\pi}{|\epsilon_1 - \epsilon_2|}$. The bound on k is then given by the minimum separation between $\boldsymbol{\epsilon}$ values

$$k \geq \frac{2\pi}{\min\limits_{i \neq j \in [1,n]} |\epsilon_i - \epsilon_j|}.$$

3.2 A k-Step Control Law

We desire the sequence of k control inputs $\mathbf{u}_{[0,\ldots,k-1]}$ that will bring the system to the goal state. If $\mathcal{C}_k$ is full rank and $k > 2n$, the system is underdetermined, with an infinite number of solutions. We choose from these solutions by finding the one with minimal control effort. That is, for any starting state $\mathbf{p}_0$ and desired final state $\mathbf{p}_k$, the control sequence is derived by minimizing $\left\|\mathbf{u}_{[0,\ldots,k-1]}\right\|_2$ subject to the constraint $\mathcal{C}_k \mathbf{u}_{[0,\ldots,k-1]} = (\mathbf{p}_k - \mathbf{p}_0)$. The solution is

$$\mathbf{u}_{[0,\ldots,k-1]} = \mathcal{C}_k^\top (\mathcal{C}_k \mathcal{C}_k^\top)^{-1} (\mathbf{p}_k - \mathbf{p}_0). \tag{6}$$

In practice, the Moore-Penrose pseudoinverse results in better numerical accuracy than the inverse above (Penrose, 1955). We note that for $k = 2n$, $\mathcal{C}$ is almost always ill-conditioned, leading to very large control commands. Best results are obtained for $k \approx 4n$, as shown in Section 3.6. When $\mathcal{C}_k$ is full rank, the robots converge exactly.

Our k-step control law allows us to move the robot exactly to n (x, y) coordinates, leaving us with one additional degree of freedom. This degree of freedom is determined by the ϕ values chosen, and an optional turn at the end. There are several possibilities. We can return the robots to their original heading, Section 3.4 discusses controlling the robot's final heading, and in Section 3.5 we use this degree of freedom to control the distribution of robots along the perimeter of star-shaped sets.

3.3 Globally Asymptotic Stabilization of An Ensemble of Unicycles

A single nonholonomic unicycle that rolls without slipping has the continuous time dynamics given in Eq. (1), repeated here for clarity:

$$\dot{\mathbf{q}}_i(t) = v_i u(t) \begin{bmatrix} \cos\theta_i \\ \sin\theta_i \\ 0 \end{bmatrix} + \epsilon_i \omega(t) \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

Given $\mathbf{q}_i(0) = [x_i(0), y_i(0), \theta_i(0)]^\intercal, \mathbf{q}_{\text{goal}} \in \mathcal{Q}$, the control problem for regulating position is to find control inputs linear velocity $u(t)$ and angular velocity $\omega(t)$ such that for any $\mathbf{q}_i(0)$ and $\mathbf{q}_{\text{goal}}$,

$$\lim_{t\to\infty} \left\| \begin{bmatrix} x_i(t) \\ y_i(t) \end{bmatrix} - \begin{bmatrix} \mathbf{q}_{\text{goal},1}(t) \\ \mathbf{q}_{\text{goal},2}(t) \end{bmatrix} \right\|_2 = 0.$$

If such inputs always exist, then we say that the system is *globally asymptotically stabilizable.*

We will solve this control problem for n nonholonomic unicycles that each receive exactly the same control inputs, but each unicycle has a unique, positive constant ϵ_i that scales the turning rate. Our control policy sets the angular velocity $\omega(t)$ to a constant non-zero value, so all the unicycles rotate in place at different rates due to their unique ϵ_i values. The control policy then sets the linear velocity $u(t)$ to decrease the position error. There exist configurations at which no $u(t)$ can decrease the position error; however, we prove that at any such configuration, except the origin, the ensemble can always rotate in place until there exists some $u(t)$ that will decrease the position error.

We choose $\omega(t) = 1$ and without loss of generality set $\theta_i(0) = 0$ so that

$$\begin{aligned} \dot{x}_i(t) &= v_i u(t)\cos(\epsilon_i t) \\ \dot{y}_i(t) &= v_i u(t)\sin(\epsilon_i t). \end{aligned} \tag{7}$$

Theorem 1. *The ensemble* (13) *for* $(v_i \neq 0, \epsilon_i \neq 0, |\epsilon_i| \neq |\epsilon_j| \;\forall\; i,j)$ *is globally asymptotically stabilizable.*

Proof. We will prove the origin is globally asymptotically stabilizable by using a control-Lyapunov function (Lyapunov, translated and edited by A.T. Fuller, 1992). A suitable Lyapunov function is the mean squared distance of the ensemble from the origin:

$$\begin{aligned} V(t,\mathbf{x},\mathbf{y}) &= \sum_{i=1}^{n} \frac{1}{2nv_i}\left(x_i^2(t) + y_i^2(t)\right) \qquad (8) \\ \dot{V}(t,\mathbf{x},\mathbf{y}) &= \sum_{i=1}^{n} \frac{1}{nv_i}\left(x_i(t)\dot{x}_i(t) + y_i(t)\dot{y}_i(t)\right) \\ &= u(t)\sum_{i=1}^{n} \frac{1}{n}\left(x_i(t)\cos(\epsilon_i t) + y_i(t)\sin(\epsilon_i t)\right) \\ &= u(t)F(t,x,y) \end{aligned}$$

Here, $F(t, \mathbf{x}, \mathbf{y})$ is the summation term which is finite as long as $x_i(0)$ and $y_i(0)$ are finite. We note here that $V(t, \mathbf{x}, \mathbf{y})$ is positive definite and radially unbounded, and $V(t, \mathbf{x}, \mathbf{y}) \equiv \mathbf{0}$ only at $(\mathbf{x}, \mathbf{y}) = (\mathbf{0}, \mathbf{0})$.

3.3.1 Designing a Control Policy

To make $\dot{V}(t, \mathbf{x}, \mathbf{y})$ negative semi-definite, we choose

$$u(t) = -F(t, \mathbf{x}, \mathbf{y})$$

For such a $u(t)$,

$$\dot{V}(t, \mathbf{x}, \mathbf{y}) = -\left(F(t, \mathbf{x}, \mathbf{y})\right)^2.$$

Note here that $\dot{V}(t, \mathbf{x}, \mathbf{y}) \leq 0$, but there exists a subspace of $(\mathbf{x}(t), \mathbf{y}(t))$ such that $\dot{V}(t, \mathbf{x}, \mathbf{y}) = 0$. Because $\dot{V}(t, \mathbf{x}, \mathbf{y})$ is negative semi-definite, we can only claim stability, not asymptotic stability. To gain a proof of asymptotic stability, we will use an approach similar to that of Beauchard et al. Beauchard et al. (2010) to apply LaSalle's invariance principle (LaSalle, 1960) to this ensemble. We will proceed by showing the invariant set contains only the origin.

3.3.2 Finding the Invariant Set

Define the set S as all configurations where no $u(t)$ exists that can decrease the Lyapunov function:

$$\begin{aligned} S &= \left\{ \mathbf{x}(t), \mathbf{y}(t) \middle| \dot{V}\left(t, \mathbf{x}(t), \mathbf{y}(t)\right)) = 0 \right\} \\ &= \left\{ \mathbf{x}(t), \mathbf{y}(t) \middle| -\left(F(t, \mathbf{x}(t), \mathbf{y}(t))\right)^2 = 0 \right\} \\ &= \left\{ \mathbf{x}(t), \mathbf{y}(t) | F(t, \mathbf{x}(t), \mathbf{y}(t)) = 0 \right\}. \end{aligned}$$

Define the time the ensemble enters S as t_0, the orientation of each robot at t_0 as $\theta_0(\epsilon)$, and $t' = t - t_0$. We then define all configurations that remain identically in S as the invariant set S_{inv}. Any configuration that enters this set will never modify its position because $u(t) = -F(t, \mathbf{x}, \mathbf{y}) = 0$ for any configuration in S_{inv}. Therefore we can drop the time-dependence of $\mathbf{x}(t)$ and $\mathbf{y}(t)$:

$$S_{inv} = \left\{ \mathbf{x}, \mathbf{y} \middle| \sum_{i=1}^{n} \left(\frac{1}{n} \left(x_i \cos(\epsilon_i t' + \theta_{0i}) + y_i \sin(\epsilon_i t' + \theta_{0i}) \right) \right), \forall t' \geq 0 \right\}.$$

We can remove $\boldsymbol{\theta}_0$ with the following change of coordinates

$$\begin{bmatrix}\mathbf{x}^*\\ \mathbf{y}^*\end{bmatrix} = \begin{bmatrix}\cos(\boldsymbol{\theta}_0) & \sin(\boldsymbol{\theta}_0)\\ -\sin(\boldsymbol{\theta}_0) & \cos(\boldsymbol{\theta}_0)\end{bmatrix}\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\end{bmatrix},$$

giving the invariant set

$$S_{inv} = \left\{\mathbf{x},\mathbf{y} \middle| \sum_{i=1}^{n}\left(\frac{1}{n}\left(x_i^*\cos(\epsilon_i t') + y_i^*\sin(\epsilon_i t')\right)\right) \equiv 0, \quad \forall t' \geq 0\right\}. \tag{9}$$

We must show that no configuration except $(\mathbf{x},\mathbf{y}) \equiv (0,0)$ is in S_{inv}. By a fundamental theorem of sampling theory, the functions $\cos(\epsilon_1 t)$, $\sin(\epsilon_1 t)$, …, $\cos(\epsilon_n t)$, $\sin(\epsilon_n t)$ are orthogonal on any closed interval of length 2π if the ϵ_i values are unique (Proakis and Manolakis, 1996). Therefore, the only constant $\mathbf{x},\mathbf{y}$ values that satisfy (15) is the origin.

We have shown that V is positive-definite and radially unbounded, $\dot{V}$ is negative semi-definite, and the only invariant point where $\dot{V} = 0$ is the origin. Therefore, we conclude that under the control policy

$$\begin{aligned} u(t) &= -\frac{1}{n}\sum_{i=1}^{n}\left(x_i(t)\cos(\epsilon_i t) + y_i(t)\sin(\epsilon_i t)\right),\\ \omega(t) &= 1. \end{aligned} \tag{10}$$

the origin of the ensemble (13) is globally asymptotically stable. □

3.4 On Controlling Heading

In our previous work with continuum ensembles, we proved the heading of an infinite ensemble is not fully controllable (Becker and Bretl, 2012a). Even with a finite ensemble of robots, the set of reachable headings is much smaller than the set of all headings. We often cannot even achieve exact consensus in heading mod(2π). However, it is possible to achieve *approximate* consensus in heading. Given a $\mu > 0$, there exists a turning command ϕ such that $|\text{mod}(\theta_i, 2\pi) - \text{mod}(\theta_j, 2\pi)| < \mu$ for all robots i, j in the ensemble.

3.4.1 Exact Heading Consensus—Infinite Ensemble

Consider an infinite ensemble of robots that have a continuum of turning rates $\boldsymbol{\epsilon} \in [\epsilon_{min}, \epsilon_{max}], \epsilon_{min} \neq \epsilon_{max}$. If these robots are initialized to the same heading, it is impossible to make the robots agree in heading at any angle other than the initial heading. Let $\gamma(T) = \frac{1}{2\pi}\int_0^T \omega(t)\,\mathrm{d}t$. Then the heading of the ensemble parameterized by $\boldsymbol{\epsilon}$ at time T is $\boldsymbol{\epsilon}\gamma(T)$, and is spread along a continuum of headings from $\epsilon_{min}\gamma(T)$ to $\epsilon_{max}\gamma(T)$.

3.4.2 Exact Heading Consensus—Finite Ensemble

For illustration, consider the hands of a 12-hour clock. The hour and minute hands overlap 22 times per day, every 12/11 hours (the first crossing is at ≈1:05:27), but the hour, minute and second hands overlap only twice: midnight and noon. This overlap occurs once per Least Common Multiple (LCM) of the periods. The LCM of a set of numbers can be obtained from their prime factorizations if the set is *mutually rational*, e.g. the ratio of any two of the numbers is a rational number (Graham et al., 1990).

Theorem 2. *It is not always possible to make a finite ensemble with different turning rates exactly agree in heading.*

Proof. It is sufficient to provide an example. Consider 3 robots; a, b, c, all with different turning rates, where

- a turns at unit velocity
- b turns at 2·unit velocity
- c turns at e·unit velocity, the base of the natural logarithm

Initialize the first two unicycles in the same direction and the third offset by π. If the unicycles are commanded to turn at a fixed turning rate, there does not exist a time when they align. Here, unicycle heading at time t is t, $t2$ and $te + \pi$. Unicycles a and b coincide infinitely often at $t = k2\pi$ for $k \in \mathbb{Z}$. Robots a, c coincide infinitely often when

$$\mathrm{mod}(t, 2\pi) = \mathrm{mod}(te + \pi, 2\pi).$$

However, the ensemble a, b, and c only coincides when:

$$\mathrm{mod}(k2\pi, 2\pi) = \mathrm{mod}(k2\pi e + \pi, 2\pi)$$

Per modular arithmetic, we divide by the common term 2π:

$$\mathrm{mod}(k, 1) = \mathrm{mod}(ke + {}^1\!/\!_2, 1) \Rightarrow 0 = \mathrm{mod}(ke + {}^1\!/\!_2, 1)$$

This equality only holds when the quantity $ke + {}^1\!/\!_2 \in \mathbb{Z}$ for $k \in \mathbb{Z}$. Since e is an irrational number, this does not occur, because the product of a rational and an irrational number is always an irrational number. □

3.4.3 Approximate Heading Consensus—Finite Ensemble

If the set of turning rates is not mutually rational, we cannot reach an exact solution. Instead we search for an approximate solution. Hurwitz' Theorem (Chandrasekharan, 1968) tells us that an irrational number ϵ_i has an infinity of rational approximations $\frac{p}{q}$ which satisfy $|\epsilon_i - \frac{p}{q}| < \frac{1}{\sqrt{5}q^2}$, where p, q are integers. In our application, a finite set of robots with unique turning rates and an error bound $\mu > 0$, Hurwitz' Theorem implies we can always find a finite $T > 0$ such that the robots' alignment error is less than μ.

The example robots in Section 3.4.2 achieve approximate heading consensus at

μ	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}
$\frac{t}{2\pi}$	1	63	3×10^3	7×10^3	2×10^6	5×10^6	6×10^6	2×10^7

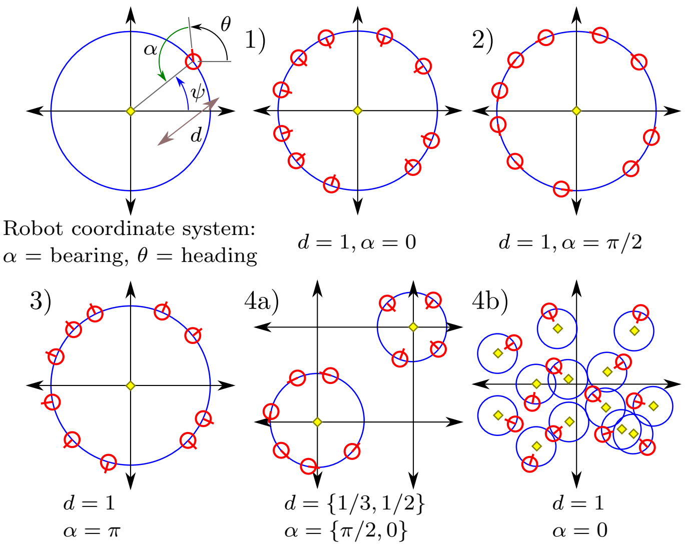
In this one example with $n = 3$ robots, the required time T for approximate convergence is approximately proportional to μ^{-1}. Because the problem of finding the least common multiple for a set of integers is NP-hard (Majewski and Havas, 1994), we predict the approximate convergence time increases superlinearly with the number of robots n.

3.5 Range and Bearing Control

We define the bearing from a robot to a fixed target as the counter-clockwise angle from the robot's heading to a vector toward the target, as shown in Fig. 6. Given a desired bearing-angle and a desired target, a finite ensemble of robots can be controlled to the perimeter of any star-shaped set around that target, as shown in Fig. 4. The set S is a *star-shaped set* if there exists q_c in S such that for all q in S the line segment from q_c to q is in S. We require a function that maps a robot's global heading θ_i to a position on the perimeter of the star-shaped set, and that the robots turn at unique rates.

Fig. 4 provides an example: 50 robots with ϵ values evenly distributed $[\frac{1}{2}, \frac{3}{2}]$ are controlled to the perimeter specified by $d(\psi) = 1 + \frac{\psi + \sin(4\psi)}{2\pi}, \psi \in [0, 2\pi]$, with bearing $\alpha = 0$. Because the ϵ_i values are mutually rational, we can achieve exact heading consensus (see Section 3.4.2). In this example the robot's headings have a period of 196π. By varying the total commanded turn $\gamma(t)$ we can modify the distribution of robots along the perimeter. Shown are $\gamma(T_1) = 50\pi$ and $\gamma(T_2) = 311$.

While our approach works on arbitrary curves, we will describe the technique by controlling the robots to the perimeter of a circle. A circular configuration best addresses our proposed tasks and the procedure is easy to



Robot coordinate system:
α = bearing, θ = heading

$d = 1, \alpha = 0$

$d = 1, \alpha = \pi/2$

$d = 1$
$\alpha = \pi$

$d = \{1/3, 1/2\}$
$\alpha = \{\pi/2, 0\}$

$d = 1$
$\alpha = 0$

Figure 6: Given n robots, where the pose of the ith robot is (x_i, y_i, θ_i), we can specify the desired bearing α and range d for each robot to at most n targets at positions (x_{ti}, y_{ti}). Let $\psi = \text{atan2}(y_i - y_{t,i}, x_i - x_{t,i})$, then the heading $\theta = \pi - \alpha + \psi$.

follow. Given n robots, we can specify the desired bearing α and range d for each robot to at most n targets. Let the pose of the ith robot be $(x_i(k), y_i(k), \theta_i(k))$.

After k moves under the motion model (3), the robot is at heading $\theta_i(k) = \theta_i(0) + \epsilon_i k\phi$. Given d, α, and the ith robot's target at $(x_{t,i}, y_{t,i})$, the desired final position is

$$\begin{bmatrix} x_i(k) \\ y_i(k) \end{bmatrix} = \begin{bmatrix} x_{t,i} \\ y_{t,i} \end{bmatrix} - \begin{bmatrix} \cos(\theta_i(0) + \epsilon_i k\phi + \alpha) \\ \sin(\theta_i(0) + \epsilon_i k\phi + \alpha) \end{bmatrix} d. \tag{11}$$

This control enables a host of configurations. Several of these are illustrated and numbered in Fig. 6. They include:

1. **Surrounding the Target** by choosing a constant d and $\alpha = 0$, the robots will finish all on a circle of radius d centered around and aimed toward the target. This configuration is suitable for monitoring a target or delivering material.

2. **Aligning Around the Target** by choosing a constant d and $\alpha = \pi/2$, the robots will form up tangent to a circle of radius d. By proper choice of ϕ and k, these robots can be distributed along the perimeter

to form a barrier around the target. This may also be a starting point for caging manipulation (Fink et al., 2007).

3. **Defending the Target** by choosing a constant d and $\alpha = \pi$, the robots will form up in a circle of radius d pointing away from the target.

4. **Mixing Configurations** up to n targets may be specified for the n robots, and each robot can be given particular values of (d, α). This may be used to surround multiple targets, form multiple layers around certain targets, or track multiple targets.

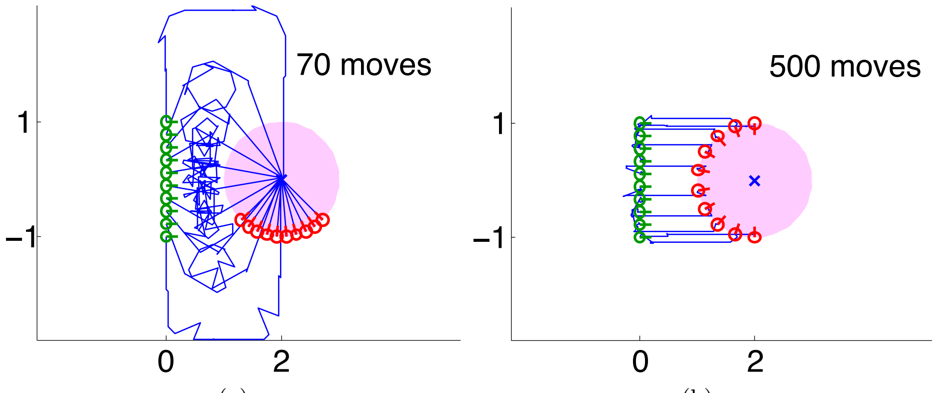
The final degree of freedom: there are $2n + 1$ controllable degrees of freedom in this system. Of these $2n$ are allocated to range and bearing (or x, y positions), leaving one input that is the integral of all turning commands. This input can be used to return the robots to their initial headings, or to permute their final spacing along the target set. Section 3.6.1 describes how permuting the final spacing can be used to search for collision-free paths. Spacing constraints are similar to the constraints on heading in Section 3.4. By similar reasoning, if the ϵ values are unique it is possible to move the robots to *approximately* the same position or *approximately* evenly space them. If the ϵ values are mutually rational, the robots can be moved to exactly the same position or be evenly distributed, as shown in Fig. 4. Even so, the set of reachable perimeter spacings is much smaller than the set of all spacings.

3.6 Open-Loop Control Results

3.6.1 Open-Loop Simulations

Each simulation starts with the ensemble of n robots initialized evenly spread with $x = 0$ and $y \in [-1, 1]$. The target is at (2,0), the commanded turn at each step $\phi = \pi/4$, and the desired range and bearing to the target is $(d, \alpha) = (1, 0)$. This setup with solutions for $k = 70$ and 500 is shown in Fig. 7.

Path Length Simulation: We examine the dependence of overall path length on the number of robots n and the number of moves k. This allows us to predict the number of moves necessary to move an ensemble of robots from start to goal configurations with a near-optimal path length. We desire short paths because under open-loop control, the true state diverges from the



(a)

(b)

Figure 7: Simulation set up. n robots are initialized evenly spread between $y = [-1, 1]$, goal is at (2,0), nominal turn $\phi = \pi/4$, desired range and bearing to the target is $(d, \alpha) = (1, 0)$, $v_i = 1$, and $\epsilon_i = \frac{1}{2} + \frac{i}{n}$. Shown above are results for $n = 10$ robots with 70 and 500 moves.

predicted state due to process noise. This noise is a function of the input commands, distance travelled, and modeling errors (Zhou and Chirikjian, 2003). To minimize state error we want to create short paths with few turns.

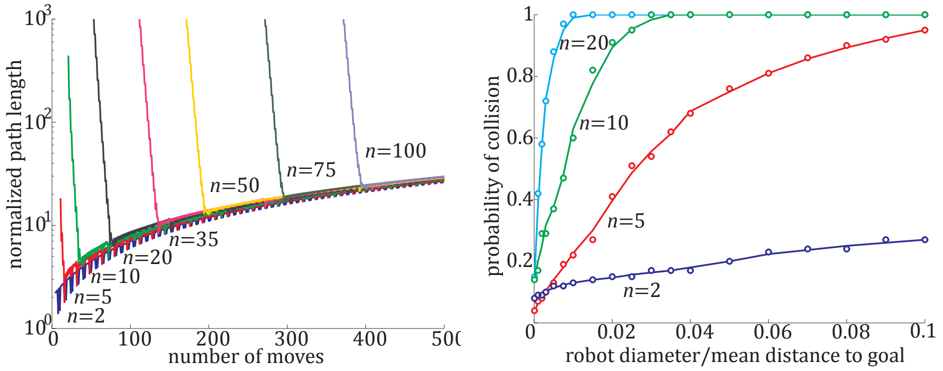
We investigated path length as a function of number of moves k for $k = [1, 500]$, and ran simulations for $n = \{2, 5, 10, 20, 35, 50, 75, 100\}$. By assigning a turning cost of 1/10 for each turn of $\phi = \pi/4$, which approximates the total distance moved for a similar turn on our hardware robots, we calculated the path length as

$$\text{path length} = k\frac{1}{10} + \sum_{i=1}^{k} |u(i)| \,.$$

To facilitate comparison, we compare using the nondimensionalized quantity

$$\text{normalized path length} = \tfrac{1}{n}\sum_{i=1}^{n} \tfrac{\text{path length}}{\text{distance}(robot_i, goal)}. \tag{12}$$

The results of these tests are shown in Fig. 8a. Note that when the number of moves is less than $2n$, the matrix $\mathcal{C}$ is not invertible. The normalized path lengths decrease from $k = 2n$ to about $k = 4n$ as $\mathcal{C}$ becomes well conditioned, then increase with k as the cost of turning dominates.



(a)

(b)

Figure 8: (a) Semi-log plot of normalized path length (12) as a function of the number of moves allowed for control law (6). Results are shown for different numbers of robots n. The same initial and target distributions are used for each test, as shown in Fig. 7. All solutions bring the robots exactly to the goal position. The paths decrease in normalized path length until $\approx 4n$ as $\mathcal{C}$ becomes well conditioned, then increase again as the cost of turning dominates. (b) Probability of a collision as a function of (robot diameter/mean distance to goal) for different numbers of robots, n. The setup in Fig. 7 is used. The probability of collision increases nonlinearly with number of robots and robot radius.

Probability of Collision Simulation: Using the same initial setup, we examined the probability of collision as a function of (robot diameter/mean distance to goal) and the number of robots. We ran simulations for $n = \{2, 5, 10, 20\}$ robots with radius values ranging from 0.001 to 0.1 units. For each radius and number of robots, we ran 100 tests by varying the turning ϕ values and number of moves k. In these tests, we checked the generated path for inter-robot collision.

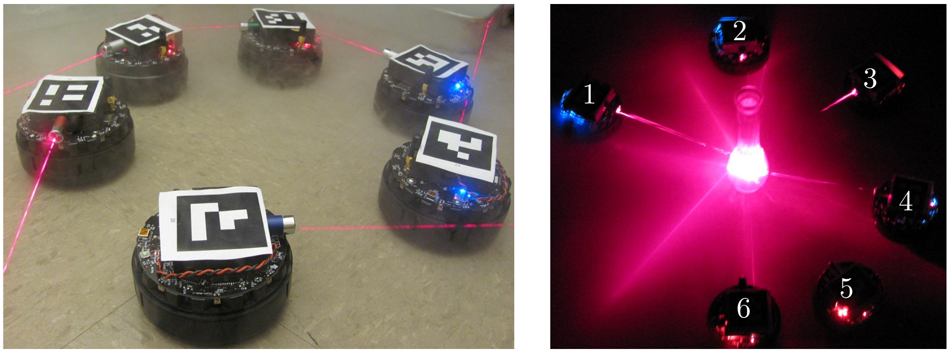
$$\text{Probability of collision} = \frac{\text{tests with collision}}{100}$$

The results of these tests are shown in Fig. 8b. The probability of collision increases nonlinearly with number of robots and robot radius. The probability that at any time, n randomly distributed robots collide grows quadratically in n. These results indicate that collisions provide a hard-limit on open-loop control for high robot densities. One method to mitigate this is to wrap (6) in a loop and check for collisions along the path. If collisions are found, new values for the turn command ϕ and the number of moves m are tried until a collision-free path is discovered or the maximum number of trials is reached. We provide code implementing this scheme in MATLAB as a free download (Becker, 2012a).

3.6.2 Open-Loop Hardware Experiments

Our differential robots (McLurkin et al., 2010) have two direct-drive wheels, and a ball caster in the back, as shown in Fig. 9. These robots are circular and can turn in place. Each robot is given a unique internal parameter ϵ_i that scales turning rate, $\epsilon_i \in [\frac{1}{2}, \frac{3}{2}]$. AprilTag fiducials(Olson, 2011) are mounted on the top of each robot and used to track robot pose. Each robot carries a laser-pointer to easily visualize the heading. We calibrated one robot using the UMBmark routine (Borenstein and Feng, 1996), then stored a unique turning rate ϵ on each robot.

For our experiments we used three r-one robots. These robots were commanded to engage a target located at (1.2,1.2) m with a desired bearing of $\alpha = \pi$ and $d = 0.3$ m. Turning rates are evenly distributed in $[\frac{1}{2}, \frac{3}{2}]$, and the initial robot positions are distributed on $x = [0, 0.2]$ m. The results of 10 hardware experiments are shown in Fig. 10. The final positions had an average distance error of 4.4 cm with standard deviation 2.8 cm and average heading error of 0.13 radians with standard deviation 0.20 radians. These small errors appear to be due to wheel slip and are within our calibration accuracy.



(a) robots with lasers illuminated by CO_2 fog (b) robots in dark, surrounding a flask

Figure 9: The r-one robots used for hardware verification. Each robot carries a laser-pointer for visualizing heading. Top-mounted AprilTag fiducials (Olson, 2011) are used for ground truth measurements. Broadcast radio commands are sent simultaneously to all robots.

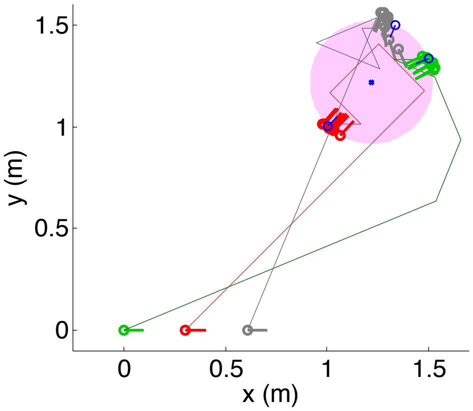


Figure 10: Hardware experiment with three robots commanded to engage a target (desired bearing $\alpha = \pi$ and $d = 0.3$ m). The target is represented by the disc centered around an 'x' at (1.2,1.2) m. Turning rates were evenly distributed in $[\frac{1}{2}, \frac{3}{2}]$, and initial robot positions distributed on $x = [0, 0.2]$ m. Shown are commanded robot paths, expected final positions, and final positions for 10 hardware experiments.

4 Closed-Loop Ensemble Control

There are several limitations to open-loop control. Pose error increases with path length, and the path length for our control law increases with the number of robots (Fig. 8a). This restricts our open-loop control solution to systems with excellent odometry or small populations of robots.

Another problem is due to robot collisions. Collisions disturb robot trajectories and may prevent robots from reaching their goal positions. The control law (6) does not account for collisions. The probability of collision increases quadratically with the number of robots.

Both problems could be alleviated by a closed-loop controller. Using global positioning sensing and a feedback control law, robots can be driven toward goal positions while rejecting disturbances. This section describes a control policy that globally asymptotically stabilizes an ensemble of unicycles controlled by uniform control inputs. Code for simulations and for generating the figures in this section is available online (Becker, 2012b).

4.1 Globally Asymptotic Stabilization of An Ensemble of Unicycles

A single nonholonomic unicycle that rolls without slipping has the continuous time dynamics given in Eq. (1), repeated here for clarity:

$$\dot{\mathbf{q}}_i(t) = v_i u(t) \begin{bmatrix} \cos\theta_i \\ \sin\theta_i \\ 0 \end{bmatrix} + \epsilon_i \omega(t) \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

Given $\mathbf{q}_i(0) = [x_i(0), y_i(0), \theta_i(0)]^\intercal, \mathbf{q}_{\text{goal}} \in \mathcal{Q}$, the control problem for regulating position is to find control inputs linear velocity $u(t)$ and angular velocity $\omega(t)$ such that for any $\mathbf{q}_i(0)$ and $\mathbf{q}_{\text{goal}}$,

$$\lim_{t\to\infty} \left\| \begin{bmatrix} x_i(t) \\ y_i(t) \end{bmatrix} - \begin{bmatrix} \mathbf{q}_{\text{goal},1}(t) \\ \mathbf{q}_{\text{goal},2}(t) \end{bmatrix} \right\|_2 = 0.$$

If such inputs always exist, then we say that the system is *globally asymptotically stabilizable.*

We will solve this control problem for n nonholonomic unicycles that each receive exactly the same control inputs, but each unicycle has a unique, positive constant ϵ_i that scales the turning rate. Our control policy sets the angular velocity $\omega(t)$ to a constant non-zero value, so all the unicycles rotate in place at different rates due to their unique ϵ_i values. The control policy then sets the linear velocity $u(t)$ to decrease the position error. There exist configurations at which no $u(t)$ can decrease the position error; however, we prove that at any such configuration, except the origin, the ensemble can always rotate in place until there exists some $u(t)$ that will decrease the position error.

We choose $\omega(t) = 1$ and without loss of generality set $\theta_i(0) = 0$ so that

$$\begin{aligned} \dot{x}_i(t) &= v_i u(t)\cos(\epsilon_i t) \\ \dot{y}_i(t) &= v_i u(t)\sin(\epsilon_i t). \end{aligned} \tag{13}$$

Theorem 3. *The ensemble* (13) *for* ($v_i \neq 0, \epsilon_i \neq 0, |\epsilon_i| \neq |\epsilon_j| \ \forall \ i,j$) *is globally asymptotically stabilizable.*

Proof. We will prove the origin is globally asymptotically stabilizable by using a control-Lyapunov function (Lyapunov, translated and edited by A.T. Fuller, 1992). A suitable Lyapunov function is the mean squared distance of the ensemble from the origin:

$$\begin{aligned} V(t,\mathbf{x},\mathbf{y}) &= \sum_{i=1}^{n} \frac{1}{2nv_i}\left(x_i^2(t) + y_i^2(t)\right) \qquad (14) \\ \dot{V}(t,\mathbf{x},\mathbf{y}) &= \sum_{i=1}^{n} \frac{1}{nv_i}\left(x_i(t)\dot{x}_i(t) + y_i(t)\dot{y}_i(t)\right) \\ &= u(t)\sum_{i=1}^{n} \frac{1}{n}\left(x_i(t)\cos(\epsilon_i t) + y_i(t)\sin(\epsilon_i t)\right) \\ &= u(t)F(t,x,y) \end{aligned}$$

Here, $F(t,\mathbf{x},\mathbf{y})$ is the summation term which is finite as long as $x_i(0)$ and $y_i(0)$ are finite. We note here that $V(t,\mathbf{x},\mathbf{y})$ is positive definite and radially unbounded, and $V(t,\mathbf{x},\mathbf{y}) \equiv \mathbf{0}$ only at $(\mathbf{x},\mathbf{y}) = (\mathbf{0},\mathbf{0})$.

4.1.1 Designing a Control Policy

To make $\dot{V}(t,\mathbf{x},\mathbf{y})$ negative semi-definite, we choose

$$u(t) = -F(t,\mathbf{x},\mathbf{y})$$

For such a $u(t)$,

$$\dot{V}(t,\mathbf{x},\mathbf{y}) = -\left(F(t,\mathbf{x},\mathbf{y})\right)^2.$$

Note here that $\dot{V}(t, \mathbf{x}, \mathbf{y}) \leq 0$, but there exists a subspace of $(\mathbf{x}(t), \mathbf{y}(t))$ such that $\dot{V}(t, \mathbf{x}, \mathbf{y}) = 0$. Because $\dot{V}(t, \mathbf{x}, \mathbf{y})$ is negative semi-definite, we can only claim stability, not asymptotic stability. To gain a proof of asymptotic stability, we will use an approach similar to that of Beauchard et al. Beauchard et al. (2010) to apply LaSalle's invariance principle (LaSalle, 1960) to this ensemble. We will proceed by showing the invariant set contains only the origin.

4.1.2 Finding the Invariant Set

Define the set S as all configurations where no $u(t)$ exists that can decrease the Lyapunov function:

$$\begin{aligned} S &= \left\{ \mathbf{x}(t), \mathbf{y}(t) \middle| \dot{V}\left(t, \mathbf{x}(t), \mathbf{y}(t)\right)\right) = 0 \right\} \\ &= \left\{ \mathbf{x}(t), \mathbf{y}(t) \middle| - \left(F(t, \mathbf{x}(t), \mathbf{y}(t))\right)^2 = 0 \right\} \\ &= \left\{ \mathbf{x}(t), \mathbf{y}(t) | F(t, \mathbf{x}(t), \mathbf{y}(t)) = 0 \right\}. \end{aligned}$$

Define the time the ensemble enters S as t_0, the orientation of each robot at t_0 as $\theta_0(\epsilon)$, and $t' = t - t_0$. We then define all configurations that remain identically in S as the invariant set S_{inv}. Any configuration that enters this set will never modify its position because $u(t) = -F(t, \mathbf{x}, \mathbf{y}) = 0$ for any configuration in S_{inv}. Therefore we can drop the time-dependence of $\mathbf{x}(t)$ and $\mathbf{y}(t)$:

$$S_{inv} = \left\{ \mathbf{x}, \mathbf{y} \middle| \sum_{i=1}^{n} \left(\frac{1}{n} \left(x_i \cos(\epsilon_i t' + \theta_{0i}) + y_i \sin(\epsilon_i t' + \theta_{0i}) \right) \right), \forall t' \geq 0 \right\}.$$

We can remove $\boldsymbol{\theta}_0$ with the following change of coordinates

$$\begin{bmatrix} \mathbf{x}^* \\ \mathbf{y}^* \end{bmatrix} = \begin{bmatrix} \cos\left(\boldsymbol{\theta}_0\right) & \sin\left(\boldsymbol{\theta}_0\right) \\ -\sin\left(\boldsymbol{\theta}_0\right) & \cos\left(\boldsymbol{\theta}_0\right) \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix},$$

giving the invariant set

$$S_{inv} = \left\{ \mathbf{x}, \mathbf{y} \middle| \sum_{i=1}^{n} \left(\frac{1}{n} \left(x_i^* \cos(\epsilon_i t') + y_i^* \sin(\epsilon_i t') \right) \right) \equiv 0, \quad \forall t' \geq 0 \right\}. \quad (15)$$

We must show that no configuration except $(\mathbf{x}, \mathbf{y}) \equiv (0, 0)$ is in S_{inv}. By a fundamental theorem of sampling theory, the functions $\cos(\epsilon_1 t)$, $\sin(\epsilon_1 t)$, ..., $\cos(\epsilon_n t)$, $\sin(\epsilon_n t)$ are orthogonal on any closed interval of length 2π if the ϵ_i values are unique (Proakis and Manolakis, 1996). Therefore, the only constant $\mathbf{x}, \mathbf{y}$ values that satisfy (15) is the origin.

We have shown that V is positive-definite and radially unbounded, $\dot{V}$ is negative semi-definite, and the only invariant point where $\dot{V} = 0$ is the origin. Therefore, we conclude that under the control policy

$$\begin{aligned} u(t) &= -\frac{1}{n} \sum_{i=1}^{n} \left(x_i(t) \cos(\epsilon_i t) + y_i(t) \sin(\epsilon_i t) \right), \\ \omega(t) &= 1. \end{aligned} \tag{16}$$

the origin of the ensemble (13) is globally asymptotically stable. □

4.2 Closed-Loop Implementation

In this section, we explain extensions of our control policy to unidirectional and discrete-time ensembles, and we apply a standard noise model to our ensemble. These extensions are useful for implementation of our policy.

4.2.1 Extension to Unidirectional Vehicles

Some systems, including the nanocar and scratch-drive microrobot, have unidirectional constraints on their inputs—they can only generate a positive linear velocity and can only turn in one direction. Our control law already uses unidirectional input for u_2. This can be extended to robots with minimum turning radius, e.g. (Donald et al., 2006, 2008), by redefining the robot center as the center of rotation. To handle linear velocity constraints, we modify (16) to be non-negative by setting $u_1(t) = \max(0, -F(t))$. In simulation and hardware experiments, the resulting unidirectional control policy converges about half as fast as the original control policy. Extending the global asymptotic stability result to unidirectional inputs is a promising avenue for future work.

4.2.2 Extension to Discrete-Time

The analysis in Section 4.1 used continuous time. Many real-life applications, including the micro- and nanorobots we discussed above, involve robots controlled and measured in discrete time.

To simplify implementation, we again use the discrete-time model (2) from Section 2.1, splitting each ΔT time step into two stages with piecewise constant inputs. During the first stage of round k we command the robots to turn in place ϕ, and during the second stage command the linear movement

$$u(k) = -\frac{1}{n}\sum_{i=1}^{n}\left(x_i(k)\cos(\theta_i(k)) + y_i(k)\sin(\theta_i(k))\right) \tag{17}$$

As long as ϕ meets the constraints on the sampling frequency given by (5), our globally asymptotically stable control results follow.

The control policy (17) is easy to implement, never increases the summed distance of the ensemble from the goal, and is robust to standard models of noise.

4.2.3 Applying a Standard Noise Model

To model process noise, we apply the noise model by Thrun et al. (Thrun et al., 2005, Chap. 5.4.2). This model defines each discrete-time motion as a rotation, a translation, and a second rotation. It uses the four parameters $\alpha_1, \alpha_2, \alpha_3$, and α_4 to weight the correlation of noise between rotation and translation actions. If the desired rotation, translation, and second rotation are given by $[\delta_{\text{rot1}}, \delta_{\text{trans}}, \delta_{\text{rot2}}]$, then the actual actions, after noise is applied, are given by

$$\begin{aligned}
\hat{\delta}_{\text{rot1}} &= \delta_{\text{rot1}} - \text{sample}(\alpha_1\delta^2_{\text{rot1}} + \alpha_2\delta^2_{\text{trans}}) \\
\hat{\delta}_{\text{trans}} &= \delta_{\text{trans}} - \text{sample}(\alpha_3\delta^2_{\text{trans}} + \alpha_4\delta^2_{\text{rot1}} + \alpha_4\delta^2_{\text{rot2}}) \\
\hat{\delta}_{\text{rot2}} &= \delta_{\text{rot2}} - \text{sample}(\alpha_1\delta^2_{\text{rot2}} + \alpha_2\delta^2_{\text{trans}}),
\end{aligned} \tag{18}$$

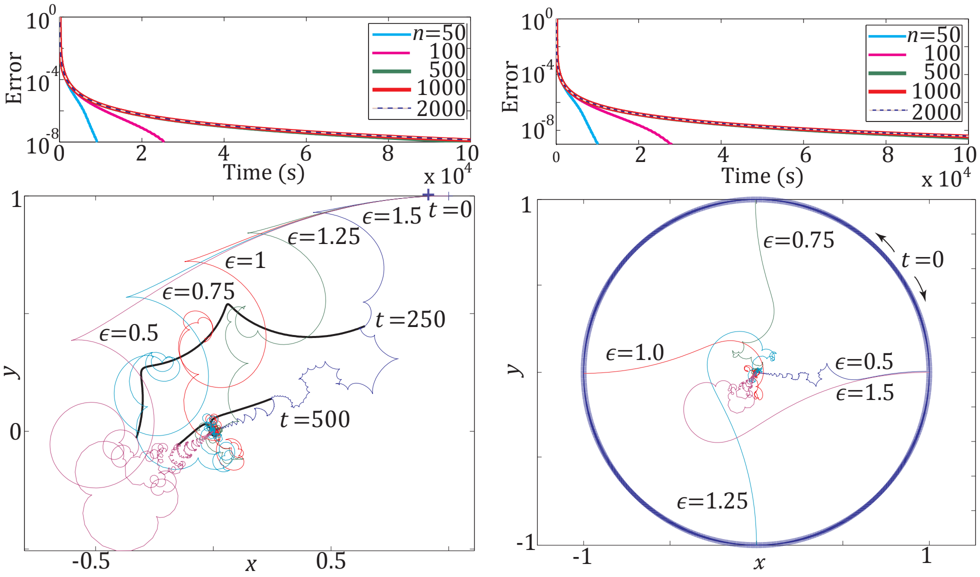
where sample(σ^2) generates a random sample from the zero-centered normal distribution with variance σ^2. We use this noise model for all discrete-time simulations.

4.3 Closed-Loop Simulation Results

Here, we present our simulation methodology and results for both continuous- and discrete-time simulations.

4.3.1 Continuous Time

We implemented a ensemble with control policy (16) in Matlab to simulate $n = \{50, 100, 500, 1000, 2000\}$ robots in continuous time for two different test cases. For these tests $v_i = \epsilon_i = \frac{1}{2} + \frac{1}{n}i$.



(a) Point to Point

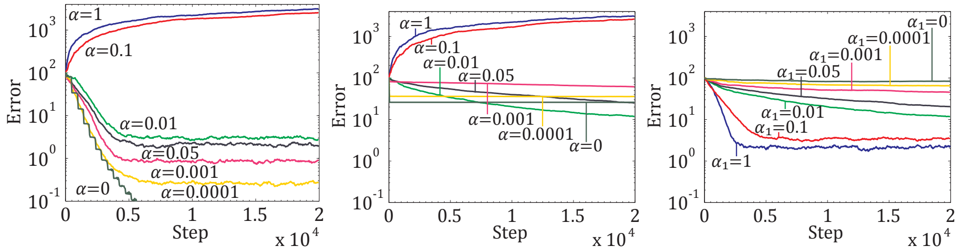
(b) Path to Point

Figure 11: Continuous time simulations of n robots, with $\epsilon \in [0.5, 1.5]$ using control policy (16) and $u_2(t) = \cos(\sqrt{t})$. Simulations were run with increasing numbers of robots. Simulations with $n \geq 500$ achieved the same error, as shown in the top plots. State trajectories of the ensemble are shown in the bottom plots. Lines show the path followed for five particular values of ϵ. Thick black lines show the entire ensemble at instants of time. (a) robots initialized to $(1, 1)$ and steered to $(0, 0)$. (b) robots initially evenly distributed about the unit circle and steered to $(0, 0)$

Point to Point Robots are initialized to $[x_i, y_i, \theta_i] = [1, 1, 0]$ and steered to the origin. Results are shown in Fig. 11a.

Path to Point Robots are initialized to $\theta_i = 2\pi\frac{i}{n}$, $[x_i, y_i] = [\cos(\theta_i, \sin(\theta_i)]$, a circle of radius 1, and steered to the origin. Results are shown in Fig. 11b.

From these simulations, we see that under our control policy, the error converges asymptotically to zero. Additionally, the Lyapunov function evolution and state trajectories for $n = 1000$ and 2000 are identical, suggesting that this level of discretization accurately represents continuum ensemble $(n = \infty)$ kinematics.



(a) Different Wheel Sizes (b) Identical Robots (c) Varying α_1, $\alpha_{2,3,4} = 0.01$

Figure 12: Convergence of the position error for discrete-time, finite collections of 120 robots simulated under a standard noise model (18). In (a) the wheel sizes $\epsilon \in [0.5, 1.5]$, while in (b) and (c), all ϵ are set to 1. (a) and (b) show different levels of noise parametrized by α; all α are equal, while (c) shows that focusing the noise in the rotation (α_1) improves convergence with identical robots.

4.3.2 Discrete Time

We simulated a discrete-time collection of 120 robots under various levels of process noise parameters with both differing and identical values of ϵ. Sample trajectories are shown in Fig. 13. We explored three different cases:

Different ϵ Values Simulating with differing ϵ, we found that with no process noise, the position error of our robot collection converged to zero error. When the noise model (18) was applied, the error converged to a non-zero value for small values of process noise, and diverged for large values, as shown in Fig. 12a.

Identical Robots When all 120 robots are identical, the smallest position error is achieved within a specific intermediate range of process noise values. Large α values caused the error to diverge, while small α values led to very slow convergence. This result is shown in Fig. 12b.

Effect of Rotational Noise Again with identical robots, we held the translational and cross-term noise at 0.01, a value which converged quickly in the previous simulation, and varied the rotational noise, α_1. Convergence rate increased with α_1, up to a limit of approximately $\alpha_1 = 1$. This result is shown in Fig. 12c.

These results show that process noise is necessary for a finite collection of identical robots to be controllable. We believe this is a subset of a larger

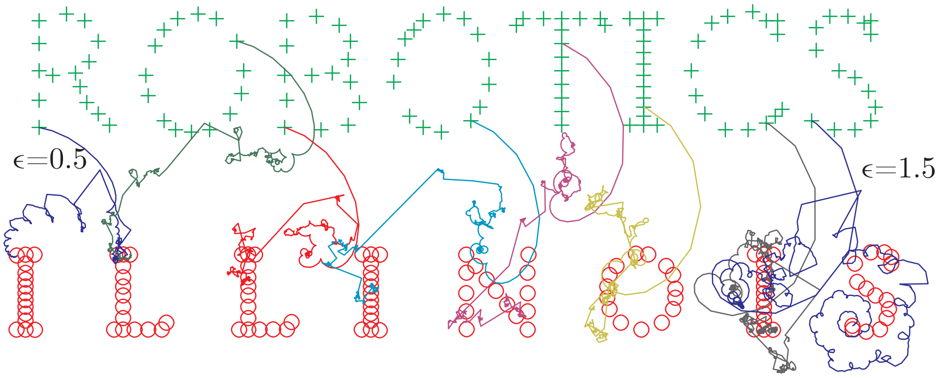


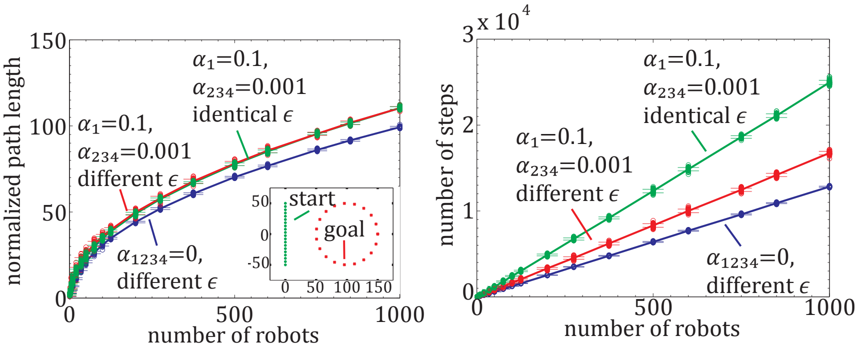
Figure 13: Simulation results from applying the control policy from (17) for 120 robots with unicycle kinematics. Wheel size (ϵ) was evenly distributed from 0.5 to 1.5. The plot shows the the starting '+' and ending '○' positions along with 8 selected state trajectories (see also Extension 1).

class of problems for which noise is beneficial, or even necessary, for stability and control. In particular, these results suggest that identical robots with uniform inputs should be designed with large rotational, but small translational and cross-term noise.

4.3.3 Convergence Time as a Function of Population Size

Using the same discrete-time model as Section 4.3.2, n=1 to 1000 robots were simulated under control law (17). The n robots were initialized evenly spread between $x = 0, y = [-50, 50]$, goal positions were equally spaced about a radius 50 circle centered at (100,0). Three scenarios were simulated, the first two with different wheel sizes: $\epsilon_i = \frac{1}{2} + \frac{i}{n}$, and the third with identical wheel sizes. The first test was noiseless ($\alpha_{1,2,3,4} = 0$), while the second and third tests had noise values $\alpha_1 = 0.1, \alpha_{2,3,4} = 0.001$. The systems were simulated until the error was below 10. The resulting normalized path lengths and total number of steps are shown in Fig. 14. The number of steps required grows roughly linearly in the number of robots n, having coefficients [13, 17, 25]. The normalized path length is shown with a linear fit in $\sqrt{n}$, having coefficients [3.1, 3.4, 3.5].

As with the open-loop controller simulations in Section 3.6, the normalized path length required grows sublinearly in the number of robots n, but the number of steps required grows linearly with n. The closed-loop controller requires more steps than the open-loop controller and ends with nonzero error. This inefficiency is caused because the closed-loop controller greedily follows the gradient, rather than solving for an optimal sequence.



(a) Path Length

(b) Number of Steps

Figure 14: Simulation results for convergence to a desired position as a function of population size under closed-loop control law (17). Inset to (a) shows starting and goal configurations. 50 trials were run for each configuration. A trend line connects the mean values and error bars show ±1 standard deviation.

Figure 14b shows the ramifications of uniform inputs compared to individually communicating with addressable robots. The path length is roughly $3.5\sqrt{n}$ longer than an addressable scheme. The total number of input commands is also greater. Uniform inputs requires $\approx 20n$ broadcast communication messages, compared to exactly $2n$ messages with addressable robots.

4.4 Closed-Loop Hardware Experiments

Here, we describe our hardware system and explain our experimental procedures and results.

4.4.1 Differential-Drive Robots

Our differential robots have two large direct-drive wheels in the back, and a free-wheeling ball caster in the front, as shown in Fig. 1. In the experiments shown in this paper, we use wheels with diameters in the set $\{102, 108, 127, 152\}$mm. Trials with approximately identical-size wheels all used 102±0.5 mm wheels.

4.4.2 System Overview

Our robots are commanded to either move linearly or turn in place in units of encoder ticks. These commands are broadcast over 900MHz radio using an AeroComm 4490 card.

Four to five tracking dots are fixed to the top of each robot. Position and orientation data for each vehicle are uniquely measured by an 18-camera NaturalPoint OptiTrack system with reported sub-millimeter accuracy. A MATLAB program computes the control policy (17) and sends the global control signal.

This setup is intended to serve as a scale model for micro and nanorobotic systems with uniform inputs, but care must be taken with broadcast radio messages. For instance, during a practice hardware test we noticed exceptionally fast convergence, and traced this to poor placement of the transmitting antenna. Due to interference, robots would occasionally not receive turn-in-place commands. This effectively introduced large process noise on the rotation and led to fast convergence, similar to that shown in Fig. 12c.

4.4.3 Online Calibration

Changing the wheel size of a differential-drive robot scales both the linear and angular turning rate, i.e. $v_i = \epsilon_i$. Surprisingly, calibration is not necessary for successful implementation of the controller. To see why, first note that the control policies (16) and (17) do not require v_i or ϵ_i values. These control policies greedily follow the gradient, so as long as the v_i and ϵ_i values have the correct signs, the control law will decrease the position error. However, in practice the policy

$$u(k) = -\frac{1}{n}\sum_{i=1}^{n}\frac{1}{\epsilon_i}\left(x_i(k)\cos(\theta_i(t)) + y_k(t)\sin(\theta_i(k))\right)$$

with the correct ϵ_i values results in faster convergence. In our hardware experiments, for every translation command $u(k)$, we record beginning and ending positions to calculate d_i, the distance traveled, and update each ϵ_i value according to the following rule

$$\epsilon_i(k+1) = \epsilon_i(k) + K\frac{|u(k)|}{M}\left(\frac{d_i}{|u(k)|} - \epsilon_i(k)\right). \tag{19}$$

K is the weighting we give new measurements of ϵ_i, and M is the maximum possible distance we may command the robot to move. For the experiments shown here $K = 0.1$ and $M = 0.7$.

4.4.4 Experiments

We conducted a series of experiments to show that our control policy converges in a real system. We show results for unique wheel sizes with online

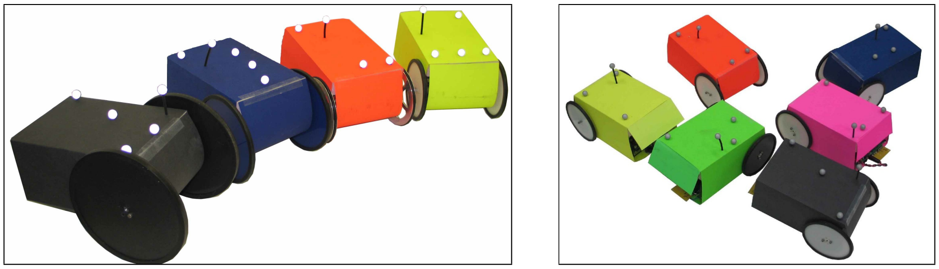


Figure 15: Four differential-drive robots with wheel diameters in the set $\{102, 108, 127, 152\}$ mm (left) and robots with 102 mm wheels (right). Each robot receives the same broadcast control signal, but the different wheel sizes scale the commanded linear and angular velocities. Robots courtesy of College of Engineering Control Systems Laboratory (Block, 2012).

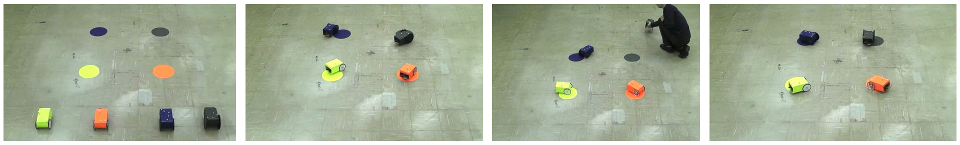


Figure 16: Photographs from hardware experiment steering four differential-drive robots with different wheel sizes. The robots are initialized in a straight line and all receive the same control input from a wireless signal. A motion capture system is used for feedback to steer the four robots to the colored targets. In the third frame a disturbance is injected by moving a single robot away from its target.

calibration, for unique wheel sizes without online calibration, and for approximately identical wheels. We conclude with a study on convergence time and steady-state error as a function of the number of robots.

Unique Wheel Sizes with Online Calibration: Initially, each of the robots in Fig. 15 was assumed to have $\epsilon_i = 1$, and the actual values of ϵ_i were learned through online calibration. The robots were successfully commanded from a horizontal line, to a box formation, to a vertical line, and finally to a tight box formation. Figure 16 shows frames from a video of this experiment. The results in Fig. 17 show convergence both in position and in ϵ_i values. Online calibration requires persistent excitation, so convergence slows as the robots approach their targets.

Unique Wheel Sizes without Calibration: It is not necessary to know or to learn the ϵ_i values. For this entire experiment ϵ_i was set to 1. The four robots in Fig. 15 were successfully commanded from a horizontal line to a box formation, and then to a vertical line. For each formation, error converged to less than half a meter, as shown in Fig. 18.

Approximately Identical Wheel Sizes: Even with approximately identical ϵ values, a collection of robots is still controllable due to process noise. The robots in Fig. 15 were fitted with approximately identical wheels. Fig. 19 shows successful convergence results of four robots with approximately identical wheel sizes commanded to the same formations as the previous experiment.

Convergence for varying numbers of robots: Our control law extends to large numbers of robots, but convergence time increases with population size. Figure 21 plots mean error as a function of time for n=1, 2, 5, 12, and 15 differential-drive robots with approximately identical wheel sizes being directed to goal positions in a regular 0.2×0.2 m grid. As shown in Fig. 20, all robots were initialized in a clump 1 meter away from the grid positions in an enclosed 2×1.5 m workspace. Heading error was artificially increased by each robot adding independent and identical turning error uniformly randomly distributed on $[-\pi/4, \pi/4]$. Robots were controlled until they achieved a steady-state error. The steady-state errors were [3, 14, 12, 24, 34] mm. In each case the position error exponentially decreases with time, but steady-state error and convergence time increase with the number of robots. The time to converge within distances $\geq$ 0.2 m of the targets is roughly linear in the number of robots n, with approximate rates of 1.8n to converge within 0.2 m and 0.34n to converge within 0.5 m. For tighter convergence the convergence time grows superlinearly, with approximate rates of 0.3n^2 to converge within 0.1 m and 0.6n^2 to converge within 0.05 m. See Extension 1 for a video of 12 r-one robots converging from one formation to a second formation.

4.4.5 Applications Enabled by Position Control

The ability to control position enables many tasks. For example, *robot aggregation* collects all the robots to one position; this primitive operation could be useful for alignment of micro- and nanorobots. To achieve aggregation, at each control step the goal position of each robot is set to the mean position of the ensemble.

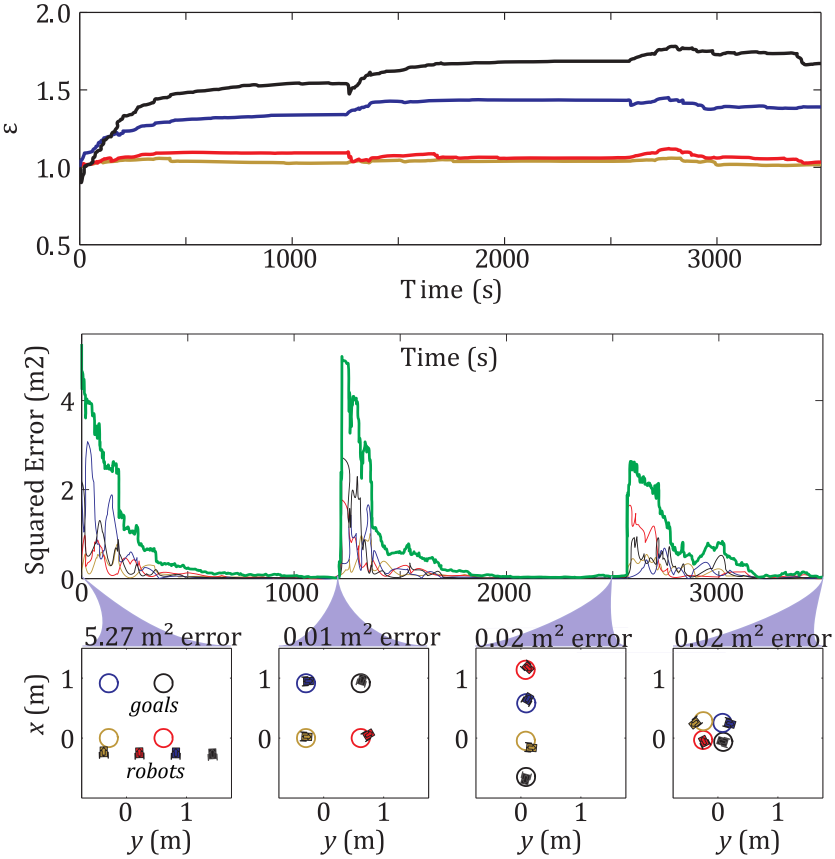


Figure 17: Hardware experiment with unique wheel sizes and online calibration. The top plot shows ϵ values estimated by online calibration. The bottom plot shows the summed distance error as the robots were steered through the sequence of formations shown.

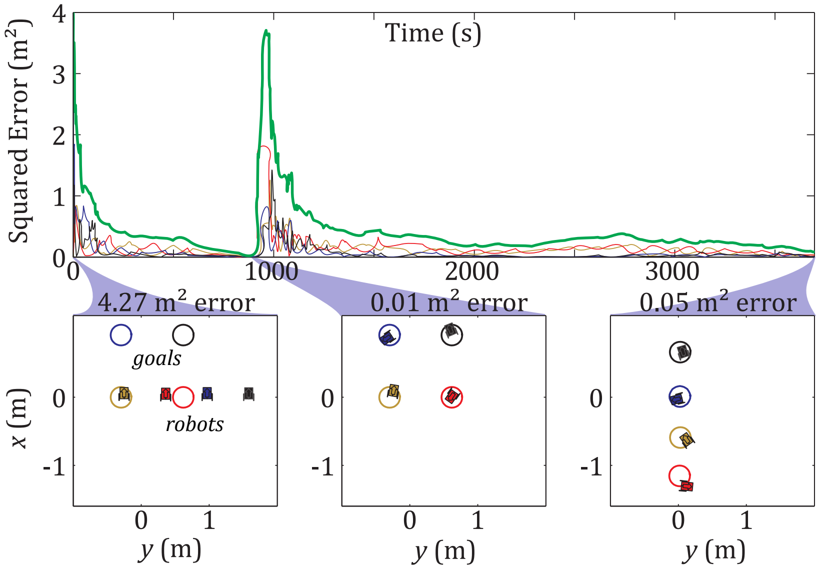


Figure 18: Hardware experiment with unique wheel sizes and no calibration. The plot shows the summed distance error as the robots were steered through the sequence of formations shown.

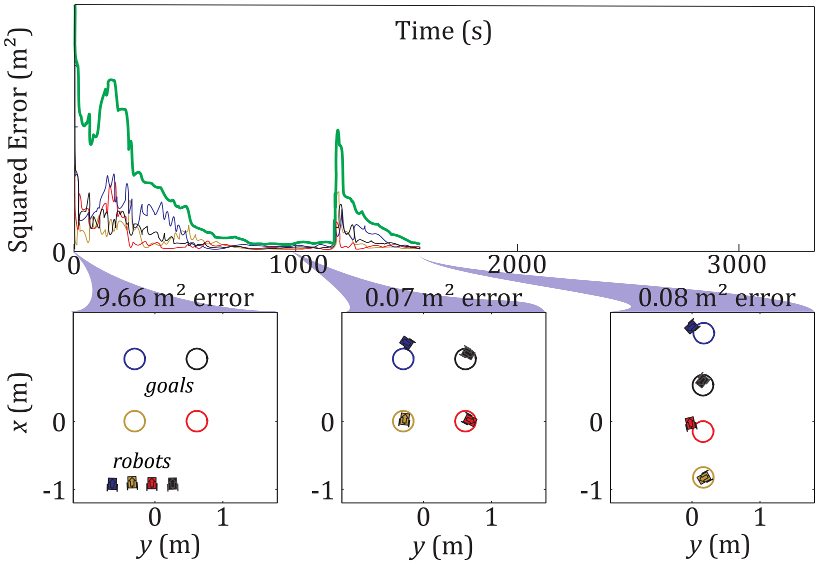
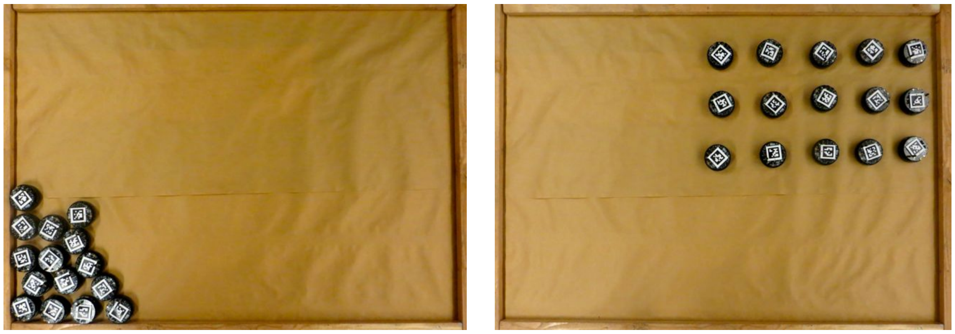


Figure 19: Hardware experiment with approximately identical wheel sizes. The plot shows the summed distance error as the robots were steered through the sequence of formations shown.



Initial configuration Goal positions

Figure 20: Measuring time for n differential-drive robots to converge to goal positions. Here $n = 15$. The workspace is 2×1.5 m, the robots are 0.1 m in diameter, and the goal positions are a regular 0.2×0.2 m grid.

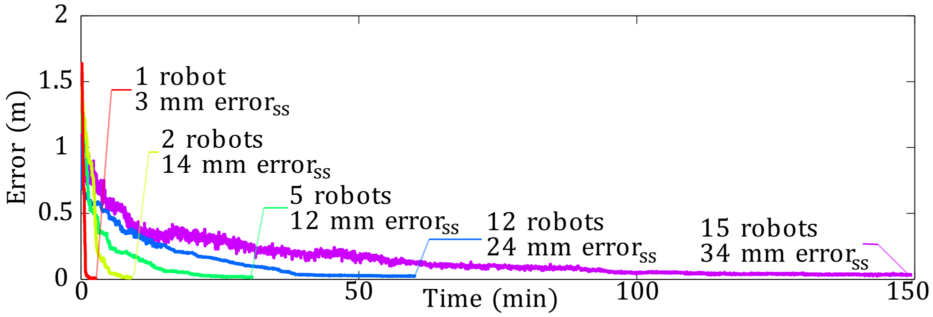


Figure 21: Hardware experiment with robots shown in Fig. 20. The goal for this experiment is to control all the robots to position them into a uniform grid. Mean error as a function of time for 1, 2, 5, 12, and 15 robots being directed to goal positions in a regular 0.2×0.2 m grid. All robots were initialized in a clump 1 meter away from the grid positions in an enclosed 2×1.5 m workspace. Robots were controlled until they reached a steady-state error.

Other tasks include forming subgroups, path- and trajectory-following, dispersion, pursuit/avoidance, manipulation, and assembly. Each can be implemented by a suitable selection of time-varying target locations in (17). See (Onyuksel, 2012, Chap. 3.6) for an implementation of trajectory tracking.

Obstacle and collision avoidance can be accomplished by adding a potential field term to the control policy (16) as in (Choset et al., 2005, Chap. 4). See (Onyuksel, 2012, Chap. 3.5) for an implementation of this obstacle avoidance method.

5 Conclusion

In this paper we investigated ensembles of nonholonomic unicycles that share a uniform control input. We first examined open-loop position and heading control of an ensemble of nonholonomic unicycles. We provided a control policy to steer n robots with unique turning rates to desired range and bearing values in a finite number of steps. This control policy was validated in simulation and in hardware experiments.

Open-loop control is rarely satisfactory due to model and process noise. Through Lyapunov analysis, we derived a globally asymptotic stabilizing controller for an ensemble of unicycles in continuous and in discrete time. In simulation, we showed that a discrete-time ensemble of unicycles converges asymptotically and rejects disturbances from a standard noise model. In hardware experiments, we demonstrated online calibration which learned the unknown parameter for each robot. These experiments led to surprising results that *1)* our controller still works when all wheel sizes are incorrectly specified and *2)* for certain levels of process noise our controller works even when all wheel sizes are the same.

This work shows that an ensemble of unicycles with uniform inputs to all robots can be regulated to arbitrary positions and reject disturbances from a standard noise model. The analysis suggests that micro- and nanorobots with uniform inputs should be designed with large rotational, but small translational process noise.

With both open- and closed-loop control, convergence time grows as a function of the number of robots. This growth makes these methods most suitable for 10s of robots. Future work should investigate how to efficiently control 100s to 1000s of nonholonomic unicycles simultaneously, and develop a theory of ensemble manipulation.

Finally, many micro and nanoscale robot systems have uniform inputs, but other motion constraints. In particular, many systems such as helical swimmers (Zhang et al., 2009b,c,a) and magnetized Tetrahymena pyriformis cells (Ou et al., 2013) move in the same direction with different speeds. See (Becker et al., 2013b) for an example of how control methods in this paper can be modified for these classes of systems.

Acknowledgements

The authors thank Dan Block, Golnaz Habibi, Seoung Kyou Lee, Nelson Chen, and Quillan Kaseman for help with the hardware experiments.

Funding

This work was supported by the National Science Foundation [grant numbers CPS-0931871, CPS-1035716, CMMI-0956362].

Appendix A: Index to Multimedia Extensions

The multimedia extension to this article is listed in Table 1, and is available at: http://www.ijrr.org.

Extension	Type	Description
1	Video	Hardware experiment with 12 differential-drive robots, all commanded by the same broadcast control signal. Robots move from a rectangular configuration to form the letter ‘R’. Next, 120 simulated robots move from “ROBOTICS” to form “IJRR” despite IID perturbation.

Table 1: Multimedia Extensions

References

Alex Ansari and T.D. Murphey. Minimal parametric sensitivity trajectories for nonlinear systems. In *American Controls Conference (ACC)*, 2013.

J.D. Badjić, V. Balzani, A. Credi, S. Silvi, and J.F. Stoddart. A molecular elevator. *Science*, 303(5665):1845–1849 0036–8075, 2004.

Karine Beauchard, Jean-Michel Coron, and Pierre Rouchon. Time-periodic feedback stabilization for an ensemble of half-spin systems. In *IFAC Sym. Nonlin. Cont. Sys.*, Bologna: Italy, September 2010.

Aaron Becker. "Range and Bearing Control of an Ensemble of Robots", September 2012a. URL `http://www.mathworks.com/matlabcentral/fileexchange/38190`.

Aaron Becker. "Feedback Control of Many Differential-Drive Robots with Uniform Control Inputs." MATLAB Central File Exchange, September 2012b. URL `http://www.mathworks.com/matlabcentral/fileexchange/38369`.

Aaron Becker and Timothy Bretl. Approximate steering of a unicycle under bounded model perturbation using ensemble control. *IEEE Trans. Robot.*, 28(3):580–591, June 2012a.

Aaron Becker and Timothy Bretl. Approximate steering of a plate-ball system under bounded model perturbation using ensemble control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, October 2012b.

Aaron Becker and James McLurkin. Exact range and bearing control of many differential-drive robots with uniform control inputs. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3338–3343, November 2013.

Aaron Becker, Cem Onyuksel, and Timothy Bretl. Feedback control of many differential-drive robots with uniform control inputs. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, October 2012.

Aaron Becker, Golnaz Habibi, Justin Werfel, Michael Rubenstein, and J. McLurkin. Massive uniform manipulation: Controlling large populations of simple robots with a common input signal. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 520–527, November 2013a.

Aaron Becker, Yan Ou, Paul Kim, MinJun Kim, and Agung Julius. Feedback control of many magnetized tetrahymena pyriformis cells by exploiting phase inhomogeneity. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3317–3323, November 2013b.

Daniel Block. The segboy, June 2012. URL http://coecsl.ece.illinois.edu/projects.html.

J. Borenstein and Liqiang Feng. Measurement and correction of systematic odometry errors in mobile robots. *IEEE Trans. Robot. Autom.*, 12(6): 869–880, Dec 1996.

R. W. Brockett and Navin Khaneja. On the control of quantum ensembles. In Theodore Djaferis and Irvin Schick, editors, *System Theory: Modeling, Analysis and Control*. Kluwer Academic Publishers, 1999.

K. Chandrasekharan. *An Introduction to Analytic Number Theory*, page 23. Springer-Verlag, Berlin, 1968.

Pinn-Tsong Chiang, Johannes Mielke, Jazmin Godoy, Jason M. Guerrero, Lawrence B. Alemany, Carlos J. Villagómez, Alex Saywell, Leonhard Grill, and James M. Tour. Toward a light-driven motorized nanocar: Synthesis and initial imaging of single molecules. *ACS Nano*, 6(1):592–597, February 2011. doi: 10.1021/nn203969b.

Howie Choset, Kevin Lynch, Seth Hutchinson, George Kantor, Wolfram Burgard, Lydia Kavraki, and Sebastian Thrun. *Principles of Robot Motion*. The MIT Press, 2005.

Dominic de Lanauze, Ouajdi Felfoul, Jean-Philippe Turcot, Mahmood Mohammadi, and Sylvain Martel. Three-dimensional remote aggregation and steering of magnetotactic bacteria microrobots for drug delivery applications. *The International Journal of Robotics Research*, 11 2013. URL http://ijr.sagepub.com/content/early/2013/11/11/0278364913500543.

E. Diller, S Floyd, C. Pawashe, and M. Sitti. Control of multiple heterogeneous magnetic microrobots in two dimensions on nonspecialized surfaces. *IEEE Trans. Robot.*, 28(1):172–182, February 2012.

Eric Diller, Joshua Giltinan, and Metin Sitti. Independent control of multiple magnetic microrobots in three dimensions. *The International Journal of Robotics Research*, 32(5):614–631, 2013. URL http://ijr.sagepub.com/content/32/5/614.abstract.

B.R. Donald, C.G. Levey, C.D. McGray, I. Paprotny, and D. Rus. An untethered, electrostatic, globally controllable MEMS micro-robot. *J. of MEMS*, 15(1):1–15, February 2006. ISSN 1057-7157.

B.R. Donald, C.G. Levey, and I. Paprotny. Planar microassembly by parallel actuation of MEMS microrobots. *J. of MEMS*, 17(4):789–808, August 2008.

Bruce R Donald, Christopher G Levey, Igor Paprotny, and Daniela Rus. Planning and control for microassembly of structures composed of stress-engineered mems microrobots. *The International Journal of Robotics Research*, 32(2):218–246, 2013. URL `http://ijr.sagepub.com/content/32/2/218.abstract`.

Albert Einstein. *Investigations on the Theory of the Brownian Movement*. Courier Dover Publications, 1956.

J. Fink, N. Michael, and V. Kumar. Composition of vector fields for multi-robot manipulation via caging. In *Robotics Science and Systems*, Atlanta, GA, June 2007.

Steven Floyd, Eric Diller, Chytra Pawashe, and Metin Sitti. Control methodologies for a heterogeneous group of untethered magnetic micro-robots. *Int. J. Robot. Res.*, 30(13):1553–1565, November 2011.

Kevin J. Gaston and Tim M. Blackburn. How many birds are there? *Biodiversity and Conservation*, 6(4):615–625, April 1997. doi: 10.1023/A:1018341530497.

Ambarish Ghosh and Peer Fischer. Controlled propulsion of artificial magnetic nanostructured propellers. *Nano Letters*, 9(6):2243–2245, 2011/10/23 2009. doi: 10.1021/nl900186w. URL `http://dx.doi.org/10.1021/nl900186w`.

R. L. Graham, D. E. Knuth, and O. Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Addison-Wesley, Reading, MA, 1990.

Takeshi Hasegawa, Naoko Ogawa, Hiromasa Oku, and Masatoshi Ishikawa. A new framework for microrobotic control of motile cells based on high-speed tracking and focusing. In *IEEE Int. Conf. Rob. Aut.*, pages 3964–3969, May 2008.

Wuming Jing, N. Pagano, and D.J. Cappelleri. A tumbling magnetic microrobot with flexible operating modes. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 5514–5519, May 2013. doi: 10.1109/ICRA.2013.6631368.

Navin Khaneja. *Geometric Control in Classical and Quantum Systems*. PhD thesis, Harvard University, 2000.

J. P. LaSalle. Some extensions of Liapunov's second method. *IRE Transactions on Circuit Theory*, CT(7):520–527, December 1960.

William S. Levine, editor. *The Control Handbook*, chapter 25.3 Discrete-Time Linear Time-Varying Systems, pages 459–463. CRC Press, Inc., United States of America, 1996.

Jr-Shin Li. Ensemble control of finite-dimensional time-varying linear systems. *IEEE Trans. Autom. Control*, 56(2):345–357, February 2011.

Jr-Shin Li and N. Khaneja. Ensemble control of Bloch equations. *IEEE Trans. Autom. Control*, 54(3):528–536, March 2009. doi: 10.1109/TAC.2009.2012983.

Pasquale Lucibello and Giuseppe Oriolo. Robust stabilization via iterative state steering with an application to chained-form systems. *Automatica*, 37:71–79, 2001.

Aleksandr Mikhailovich Lyapunov, translated and edited by A.T. Fuller. *The General Problem of the Stability of Motion*. Tayor & Francis, London, 1992.

Bohdan S. Majewski and George Havas. The complexity of greatest common divisor computations. In LeonardM. Adleman and Ming-Deh Huang, editors, *Algorithmic Number Theory*, volume 877 of *Lecture Notes in Computer Science*, pages 184–193. Springer Berlin Heidelberg, 1994. ISBN 978-3-540-58691-3. doi: 10.1007/3-540-58691-1_56. URL `http://dx.doi.org/10.1007/3-540-58691-1_56`.

J. McLurkin, A. Lynch, S. Rixner, T. Barr, A. Chou, K. Foster, and S. Bilstein. A low-cost multi-robot system for research, teaching, and outreach. *Distributed Autonomous Robotic Systems*, pages 597–609, 2010.

A. Minett, J. Fràysse, G. Gang, G.T. Kim, and S. Roth. Nanotube actuators for nanomechanics. *Current Applied Physics*, 2(1):61–64 1567–1739, 2002.

E. Olson. AprilTag: A robust and flexible visual fiducial system. In *IEEE Int. Conf. Rob. Aut.*, pages 3400–3407. IEEE, May 2011.

Cem Onyuksel. Feedback control of many differential-drive robots with uniform control inputs. Master's thesis, University of Illinois at Urbana-Champaign, http://hdl.handle.net/2142/34414, August 2012.

Yan Ou, Dal Hyung Kim, Paul Kim, Min Jun Kim, and A. Agung Julius. Motion control of magnetized tetrahymena pyriformis cells by magnetic field with model predictive control. *Int. J. Rob. Res.*, 32(1):129–139, January 2013.

R. Penrose. A generalized inverse for matrices. *Proc. Cambridge Phil. Soc.*, 51:406–413, 1955.

Kathrin E. Peyer, Li Zhang, and Bradley J. Nelson. Bio-inspired magnetic swimming microrobots for biomedical applications. *Nanoscale*, 2013.

Michiel Plooij, Michiel de Vries, Wouter Wolfslag, and Martijn Wisse. Optimization of feedforward controllers to minimize sensitivity to model inaccuracies. In *IEEE Int. Rob. and Sys.*, 2013.

John G. Proakis and Dimitris G. Manolakis. *Digital signal processing: principles, algorithms, and applications.* Prentice Hall, Englewood Cliffs, NJ, 1996.

S. Schürle, K. E. Peyer, B. E. Kratochvil, and B. J. Nelson. Holonomic 5-DOF magnetic control of 1D nanostructures. In *IEEE Int. Conf. Rob. Aut.*, pages 1081–1086, May 2012.

C. E. Shannon. Communication in the presence of noise. *Proc. Institute of Radio Engineers, Reprint as classic paper in: Proc. IEEE, vol. 86, no. 2, (Feb. 1998)*, 37(1):10–21, January 1949.

Yong Kyun Shin, Yeongjin Kim, and Jung Kim. Automated microfluidic system for orientation control of mouse embryos. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 496–501. IEEE, 2013.

Yasuhiro Shirai, Andrew J. Osgood, Yuming Zhao, Kevin F. Kelly, and James M. Tour. Directional control in thermally driven single-molecule nanocars. *Nano Letters*, 5(11):2330–2334, February 2005. doi: 10.1021/nl051915k.

E. B. Steager, M.S. Sakar, D. H. Kim, V. Kumar, and G. J. Pappas. Electrokinetic and optical control of bacterial microrobots. *J. of Micromechanics and Microengineering*, 21(3), March 2011.

Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents).* The MIT Press, September 2005.

S. Tottori, L. Zhang, F. Qiu, K.K. Krawczyk, A. Franco-Obregón, and B. J. Nelson. Magnetic helical micromachines: Fabrication, controlled swimming, and cargo transport. *Advanced Materials*, 24(811), 2012.

Hsi-Wen Tung, Dominic R. Frutiger, Salvador Panè, and Bradley J. Nelson. Polymer-based wireless resonant magnetic microrobots. In *IEEE International Conference on Robotics and Automation*, pages 715–720, May 2012.

Panagiotis Vartholomeos, M.Reza Akhavan-Sharif, and Pierre E. Dupont. Motion planning for multiple millimeter-scale magnetic capsules in a fluid environment. In *IEEE Int. Conf. Rob. Aut.*, pages 1927–1932, May 2012.

Guillaume Vives, JungHo Kang, Kevin F. Kelly, and James M. Tour. Molecular machinery: Synthesis of a "nanodragster". *Organic Letters*, 11(24):5602–5605, 2012/08/03 2009. doi: 10.1021/ol902312m. URL `http://dx.doi.org/10.1021/ol902312m`.

Li Zhang, Jacob J. Abbott, Lixin X. Dong, Bradley E Kratochvil, Dominik J. Bell, and Bradley J. Nelson. Artificial bacterial flagella: Fabrication and magnetic control. *Applied Physics Letters*, 94(6), February 2009a. Art. No. 064107.

Li Zhang, Jacob J. Abbott, Lixin X. Dong, Bradley E Kratochvil, H. X. Zhang, Kathrin E. Peyer, and Bradley J. Nelson. Micromanipulation using artificial bacterial flagella. In *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, St. Louis, MO, USA, October 2009b.

Li Zhang, Jacob J. Abbott, Lixin X. Dong, Kathrin E. Peyer, Bradley E Kratochvil, H. X. Zhang, Christos Bergeles, and Bradley J. Nelson. Characterizing the swimming properties of artificial bacterial flagella. *Nano Letters*, 9(10):3663–3667, October 2009c. doi: 10.1021/nl901869j.

Yu Zhou and G.S. Chirikjian. Probabilistic models of dead-reckoning error in nonholonomic mobile robots. In *IEEE Int. Conf. Rob. Aut.*, volume 2, pages 1594–1599, September 2003.