

# Stochastic Swarm Control with Global Inputs

Shiva Shahrokhi and Aaron T. Becker

**Abstract**—Simple robots can do huge things. To control swarm of simple robots, we need to know how to steer them simply. One way is global control which means a single input to all the robots (or particles). Previous works show that humans can do it, so we want computers to do it automatically and efficiently.

## I. INTRODUCTION

Large populations of micro- and nanorobots are being produced in laboratories around the world, with diverse potential applications in drug delivery and construction [1]–[3]. These activities require robots that behave intelligently. Limited computation and communication rules out autonomous operation or direct control over individual units; instead we must rely on global control signals broadcast to the entire robot population. It is not always practical to gather pose information on individual robots for feedback control; the robots might be difficult or impossible to sense individually due to their size and location. However, it is often possible to sense global properties of the group, such as mean position and density. Finally, many promising applications will require direct human control, but user interfaces to thousands—or millions—of robots is a daunting human-swarm interaction (HSI) challenge.

The goal of this work is to provide a tool for investigating HSI methods through statistically significant numbers of experiments. There is currently no comprehensive understanding of user interfaces for controlling multi-robot systems with massive populations. We are particularly motivated by the sharp constraints in micro- and nanorobotic systems. For example, full-state feedback with  $10^6$  robots leads to operator overload. Similarly, the user interaction required to individually control each robot scales linearly with robot population. Instead, user interaction is often constrained to modifying a global input. This input may be nonstandard, such as the attraction/repulsion field from a scanning tunneling microscope (STM) tip.

Our previous work with over a hundred hardware robots and thousands of simulated robots [4] demonstrated that direct human control of large swarms is possible. Unfortunately, the logistical challenges of repeated experiments with over one hundred robots prevented large-scale tests.

Our goal was to test several scenarios involving large-scale human-swarm interaction (HSI), and to do so with a statistically-significant sample size. Towards this end, we created SwarmControl.net, an open-source online testing

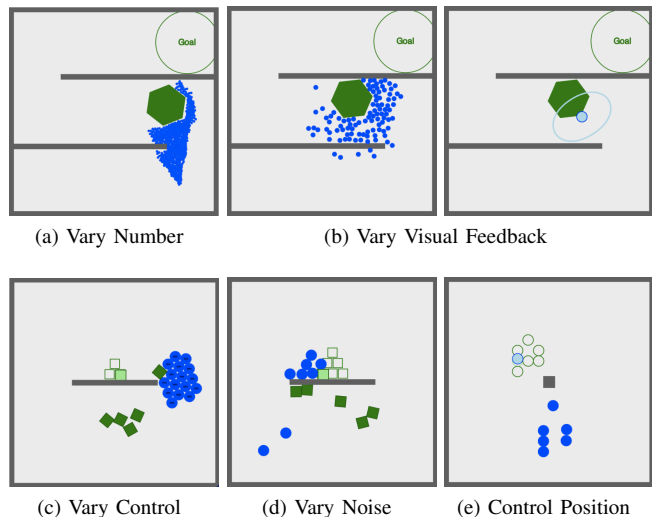


Fig. 1. Screenshots from our five online experiments controlling multi-robot systems with limited, global control. (a) Varying the number of robots from 1-500 (b) Comparing 4 levels of visual feedback (c) Comparing 3 control architectures (d) Varying noise from 0 to 200% of control authority (e) Controlling the position of 1 to 10 robots. See video overview at <http://youtu.be/HgNENj3hvEg>.

platform suitable for inexpensive deployment and data collection on a scale not yet seen in swarm robotics research. Screenshots from this platform are shown in Fig. 1. All code [5], and experimental results are posted online.

Our experiments show that numerous simple robots responding to global control inputs are directly controllable by a human operator without special training, that the visual feedback of the swarm state should be very simple in order to increase task performance, and that humans perform swarm-object manipulation faster using attractive control schemes than repulsive control schemes.

Our paper is organized as follows. After a discussion of related work in Section II, we describe our experimental methods for an online human-user experiment in Section ?? . We report the results of our experiments in Section V, discuss the lessons learned in Section ?? , and end with concluding remarks in Section VI.

## II. RELATED WORK

### A. Human-Swarm Interaction

Olson and Wood studied human *fanout*, the number of robots a single human user could control [6]. They postulated that the optimal number of robots was approximately the autonomous time divided by the interaction time required by each robot. Their sample problem involved a multi-robot search task, where users could assign goals to robots.

Their user interaction studies with simulated planar robots indicated a *fanout plateau* of about 8 robots, after which there were diminishing returns. They hypothesize that the location of this plateau is highly dependent on the underlying task, and our work indicated there are some tasks without plateaus. Their research investigated robots with 3 levels of autonomy. We use robots without autonomy, corresponding with their first-level robots.

Squire, Trafton, and Parasuraman designed experiments showing that user-interface design had a high impact on the task effectiveness and the number of robots that could be controlled simultaneously in a multi-robot task [7].

A number of user studies compare methods for controlling large swarms of simulated robots, for example [8]–[10]. These studies provide insights but are limited by cost to small user studies; have a closed-source code base; and focus on controlling intelligent, programmable agents. For instance [10] was limited to a pool of 18 participants, [8] 5, and [9] 32. Using an online testing environment, we conduct similar studies but with much larger sample sizes.

### B. Global-control of micro- and nanorobots

Small robots have been constructed with physical heterogeneity so that they respond differently to a global, broadcast control signal. Examples include *scratch-drive microrobots*, actuated and controlled by a DC voltage signal from a substrate [11], [12]; magnetic structures with different cross-sections that could be independently steered [13], [14]; *MagMite* microrobots with different resonant frequencies and a global magnetic field [15]; and magnetically controlled nanoscale helical screws constructed to stop movement at different cutoff frequencies of a global magnetic field [1], [16].

Similarly, our previous work [17], [18] focused on exploiting inhomogeneity between robots. These control algorithms theoretically apply to any number of robots—even robotic continuums—but in practice process noise cancels the differentiating effects of inhomogeneity for more than tens of robots. We desire control algorithms that extend to many thousands of robots.

### C. Three challenges for massive manipulation

While it is now possible to create many micro- and nanorobots, there remain challenges in control, sensing, and computation.

1) *Control—global inputs*: Many micro- and nanorobotic systems [1]–[3], [11]–[16], [19] rely on global inputs, where each robot receives an exact copy of the control signal. Our experiments follow this global model.

2) *Sensing—large populations*: Parallel control of  $n$  differential-drive robots in a plane requires  $3n$  state variables. Even holonomic robots require  $2n$  state variables. Numerous methods exist for measuring this state in micro- and nanorobotics. These solutions use computer vision systems to sense position and heading angle, with corresponding challenges of handling missed detections and image registration between detections and robots. These challenges

are increased at the nanoscale where sensing competes with control for communication bandwidth. We examine control when the operator has access to partial feedback, including only the first and/or second moments of a population's position, or only the convex-hull containing the robots.

3) *Computation—calculating the control law*: In our previous work the controllers required at best a summation over all the robot states [18] and at worst a matrix inversion [17]. These operations become intractable for large populations of robots. By focusing on *human* control of large robot populations, we accentuate computational difficulties because the controllers are implemented by the unaided human operator.

## III. THEORY

### A. Models

We consider holonomic robots that move in the 2D plane. We want to control position and velocity of the robots and deciding about force for reaching our desired position and velocity. So our input is going to be acceleration ( $F = ma$ ). If we show acceleration by  $a$ , velocity by  $v$  and position in  $x$  coordinate by  $p_x$  and in  $y$  coordinate by  $p_y$ , we have the following equations

$$\begin{bmatrix} \dot{p}_x = v_x \\ \dot{v}_x = a_x \end{bmatrix} \quad (1)$$

$$\begin{bmatrix} \dot{p}_y = v_y \\ \dot{v}_y = a_y \end{bmatrix} \quad (2)$$

The state-space representation of our OpenLoop controller is:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y &= Cx(t) + Du(t) \end{aligned} \quad (3)$$

where  $x(t)$  represents our states,  $u(t)$  is our input. We also have  $y$  as our output.

First we assume that we don't have noise in the system and we also have just one robot. As mentioned,  $a$  is our input and  $x$ ,  $y$ , and their velocities are our states that we want to control.

We define our states as following:

$$\begin{aligned} x_1 &= p_x \\ \dot{x}_1 &= x_2 = \dot{p}_x = v_x \\ x_3 &= p_y \\ \dot{x}_3 &= x_4 = \dot{p}_y = v_y \end{aligned} \quad (4)$$

So our state space representation is:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} u \quad (5)$$

We want to find number of states that we can control. We need to know rank of the controllability matrix  $\{B, AB, A^2B, \dots, A^{n-1}B\}$ .

$$C = \{B, AB, A^2B, \dots, A^{n-1}B\} \quad (6)$$

$$C = \left\{ \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \right\} \quad (7)$$

So we showed that we have two controllable states.

### B. Independent control is not possible

We know that we can control velocity of our robot, we want to see what happens if we had more than one robot? As we saw here,  $v_x$  is completely independent of  $v_y$ , so if we wanted to have  $n$  robots, we had  $n$  states in  $x$  axis, and  $n$  states in  $y$  axis, which were completely independent from each other. So assume we have  $n$  robots and want to control them in  $x$  axis:

$$\begin{aligned} \dot{p}_{x1} &= v_{x1} \\ \dot{v}_{x1} &= a_{x1} \\ \dot{p}_{x2} &= v_{x2} \\ \dot{v}_{x2} &= a_{x2} \\ &\vdots \\ \dot{p}_{xn} &= v_{xn} \\ \dot{v}_{xn} &= a_{xn} \end{aligned} \quad (8)$$

So our state-space representation will be:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_{2n-1} \\ \dot{x}_{2n} \end{bmatrix} = \begin{bmatrix} 0 & 1 & . & . & . & 0 & 0 \\ 0 & 0 & . & . & . & 0 & 0 \\ 0 & 0 & 0 & 1 & . & 0 & 0 \\ 0 & 0 & 0 & 0 & . & 0 & 0 \\ . & . & . & . & . & . & . \\ 0 & 0 & . & . & . & 0 & 1 \\ 0 & 0 & . & . & . & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ . \\ . \\ . \\ x_{2n-1} \\ x_{2n} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ . \\ . \\ . \\ 0 \\ 1 \end{bmatrix} a_x \quad (9)$$

If we had  $n$  robots, we had exactly the same symmetry of what we had for 1 robot. We can again control two states because we have rank two in  $C$ :

$$C = \left\{ \begin{bmatrix} 0 \\ 1 \\ . \\ . \\ . \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ . \\ . \\ . \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ . \\ . \\ . \\ 0 \\ 0 \end{bmatrix}, \dots \right\} \quad (10)$$

### C. Controlling Mean Position

So for any number of robots if we give a global command to them, we have just two controllable states in each axis. So it is obvious that we can not control position of all the robots, but what states are controllable? To answer this question we create a reduced order system that calculates average position and average velocity of the robots:

$$\begin{bmatrix} \dot{\bar{x}}_p \\ \dot{\bar{x}}_v \end{bmatrix} = \frac{1}{n} \begin{bmatrix} 0 & 1 & 0 & 1 & \dots & 0 & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ . \\ . \\ . \\ x_{2n-1} \\ x_{2n} \end{bmatrix} + \frac{1}{n} \begin{bmatrix} 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & 1 & \dots & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ . \\ . \\ . \\ 0 \\ 1 \end{bmatrix} u_x \quad (11)$$

Thus:

$$\begin{bmatrix} \dot{\bar{x}}_p \\ \dot{\bar{x}}_v \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \bar{x}_p \\ \bar{x}_v \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \quad (12)$$

We analyze  $C$  for  $y$ :

$$C = \left\{ \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right\} \quad (13)$$

This matrix again has rank two, and thus all the states are controllable. These controllable states are the average position and average velocity:

$$a = K \left[ \begin{bmatrix} \dot{\bar{x}}_p \\ \dot{\bar{x}}_v \end{bmatrix} - \begin{bmatrix} \dot{x}_{goal} \\ \dot{x}_{goal} \end{bmatrix} \right] \quad (14)$$

■

### D. Controlling the variance of many robots

As shown in section ?? above, only the mean position of a group of robots is controllable. However, there are several techniques for breaking symmetry, for example by allowing independent noise sources [20], or by using obstacles [21].

Control the variance requires being able to increase and decrease the variance. Given a large free workspace, Brownian noise is sufficient to increase the variance. A flat obstacle can be used to decrease variance. We know the standard deviation of the robots with this equation:  $\sigma$  is standard deviation:

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2} \quad (15)$$

Consider the following Control Lyapunov Function:

$$V = 1/2(\sigma_{goal} - \sigma(x))^2 \quad (16)$$

If we calculate the derivative of Lyapunov function we have:

$$\dot{V} = (\sigma_{goal} - \sigma(x))\dot{\sigma} \quad (17)$$

For not getting stuck for the goal variance, we will have two goal variances, one is the minimum variance desired which when we get to this goal, we drop decreasing variance and when the variance increases to the maximum desired

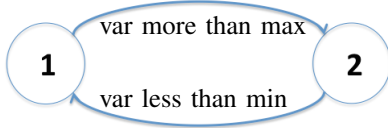


Fig. 2. Two states for controlling variances.

variance we start to decrease variance. So we use a hysteresis function to control variance:

$$switch = \begin{cases} 1 \rightarrow 2 & var > max\_var \\ 2 \rightarrow 1 & var < min\_var \end{cases} \quad (18)$$

Real systems, especially at the micro scale, are affected by unmodelled dynamics much of which can be designed by Brownian noise. To model this equation (3) must be modified as follows:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + We(t) \\ y &= Cx(t) + Du(t) \end{aligned} \quad (19)$$

where  $e(t)$  is the error in the system.

After some time, gaussian distribution shapes the outline of the robots because of the Brownian noise feature:

$$P(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2} \quad (20)$$

#### E. Controlling both the mean and the variance of many robots

When we want to control mean and variance in the same time, we should use a hysteresis control law for the variance along controlling mean. It means that we control mean position while variance is not bigger than our maximum desired variance. If variance was bigger than the maximum desired variance, we then stop controlling mean position and reduce the variance. when we reach some minimum desired variance, we go back to our mean position control.

(21)

### IV. SIMULATION

#### A. Controlling the mean position

For controlling mean position, we use a PID controller. Our control input is the force we make, goal positions are the desired positions, and we have our mean position and mean velocity. So we have:

$$\begin{aligned} control\_input_x &= K_g * (goal_x - \bar{x}) + K_d * (0 - \bar{v}_x) \\ control\_input_y &= K_g * (goal_y - \bar{y}) + K_d * (0 - \bar{v}_y) \end{aligned} \quad (22)$$

where  $K_{gain}$  is the gain value, and  $K_{derivative}$  is the derivative value. In order to find these gain values, we tried different values to find the best between them as shown in figure ???. In this experiment 100 robots were used and the maximum speed would be 3 cell per second. And with these parameters, we showed that the best values for  $K_{gain}$  is 4, and the best value for  $K_{derivative}$  is 1.

#### B. Controlling the variance

For Controlling variance, as we discussed in section ???, we use a hysteresis variance control. For increasing variance we just need to wait and maybe it would be better to go to the area with less obstacles to speed up the process. And in order to decrease the variance, we go to corners to make the variance less.

contrast controllers – as is typical with PID control laws, we can tune the response to meet desired specifications.

image showing varying Brownian noise

image showing control x variance and y-variance out of phase

#### C. Hysteresis Control of mean and variance

plot showing 1.5 cycles of mean position, and a variance goal. We might need a longer time

### V. RESULTS

The experimental section compares the results of our hysteresis based controller applied to a *block-pushing* task. In preliminary work over 1000 human users completed this task using varying levels of feedback. To our surprise, users who received the lowest amount of feedback – just the moments of the position distribution of the robot swarm – performed better than users with full state feedback.

#### A. Human-user experiment

Sensing is expensive, especially on the nanoscale. To see nanocars [3], scientists fasten molecules that fluoresce light when activated by a strong light source. Unfortunately, multiple exposures can destroy these molecules, a process called *photobleaching*. Photobleaching can be minimized by lowering the excitation light intensity, but this increases the probability of missed detections [22]. This experiment explores manipulation with varying amounts of sensing information: **full-state** sensing provides the most information by showing the position of all robots; **convex-hull** draws a convex hull around the outermost robots; **mean** provides the average position of the population; and **mean + variance** adds a confidence ellipse. Fig. 4 shows screenshots of the same robot swarm with each type of visual feedback. Full-state requires  $2n$  data points for  $n$  robots. Convex-hull requires at worst  $2n$ , but usually a smaller number. Mean requires two, and variance three, data points. Mean and mean + variance are convenient even with millions of robots. Our hypothesis predicted a steady decay in performance as the amount of visual feedback decreased.

To our surprise, our experiment indicates the opposite: players with just the mean completed the task faster than those with full-state feedback. As Fig. 3 shows, the levels of feedback arranged by increasing completion time are [mean + variance, mean, full-state, convex-hull]. Anecdotal evidence from beta-testers who played the game suggests that tracking 100 robots is overwhelming—similar to schooling phenomena that confuse predators—while working with just the mean + variance is like using a “spongy” manipulator. Our beta-testers found convex-hull feedback confusing

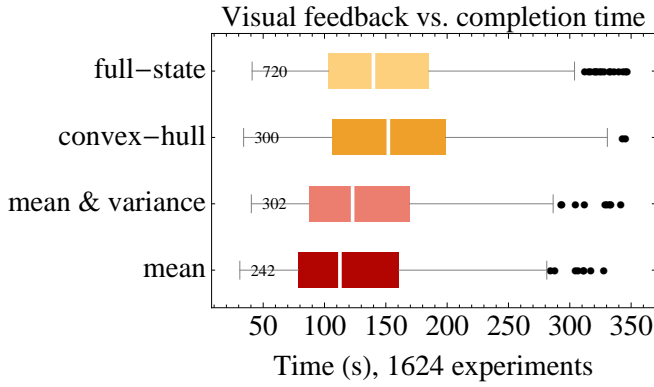


Fig. 3. Completion-time results for the four levels of visual feedback shown in Fig. 4. Surprisingly, players perform better with limited feedback—subjects with only the mean + variance outperformed all others.

and irritating. A single robot left behind an obstacle will stretch the entire hull, obscuring the majority of the swarm.

### B. Automated Block Pushing

To solve this problem, we discretized the environment, used breadth-first search to determine  $M$ , the shortest path from any point for the block to the goal, and generate a gradient map  $\nabla M$  toward the goal. The block's center of mass is at  $b$  and has radius  $r_b$ . The robots were then directed to assemble at  $b - r_b \nabla M$  to push the block toward the goal location.

TODO: write this in algorithmic form

image: show the vector field and the robots pushing a block

plot of results, comparing to humans

plot of results, varying the number of robots

images of worlds where the algorithm fails, short discussion.

## VI. CONCLUSION AND FUTURE WORK

We ...

## VII. ACKNOWLEDGEMENTS

### REFERENCES

- [1] K. E. Peyer, L. Zhang, and B. J. Nelson, "Bio-inspired magnetic swimming microrobots for biomedical applications," *Nanoscale*, 2013.
- [2] Y. Shirai, A. J. Osgood, Y. Zhao, K. F. Kelly, and J. M. Tour, "Directional control in thermally driven single-molecule nanocars," *Nano Letters*, vol. 5, no. 11, pp. 2330–2334, Feb. 2005.
- [3] P.-T. Chiang, J. Mielke, J. Godoy, J. M. Guerrero, L. B. Alemany, C. J. Villagómez, A. Saywell, L. Grill, and J. M. Tour, "Toward a light-driven motorized nanocar: Synthesis and initial imaging of single molecules," *ACS Nano*, vol. 6, no. 1, pp. 592–597, Feb. 2011.

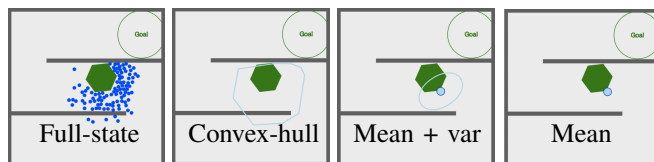


Fig. 4. Screenshots from task *Vary Visualization*. This experiment challenges players to quickly steer 100 robots (blue discs) to push an object (green hexagon) into a goal region. We record the completion time and other statistics.

- [4] A. Becker, G. Habibi, J. Werfel, M. 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)*, Nov. 2013, pp. 520–527.
- [5] C. Ertel and A. Becker. (2013, Sep.) Swarmcontrol git repository. [Online]. Available: <https://github.com/crtel/swarmmanipulate.git>
- [6] D. R. Olsen Jr and S. B. Wood, "Fan-out: Measuring human control of multiple robots," in *SIGCHI Conference on Human Factors in Computing Systems*, Vienna, Austria, Apr. 2004, pp. 231–238.
- [7] P. Squire, G. Trafton, and R. Parasuraman, "Human control of multiple unmanned vehicles: effects of interface type on execution and task switching times," in *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*, ser. HRI '06. New York, NY, USA: ACM, 2006, pp. 26–32. [Online]. Available: <http://doi.acm.org/10.1145/1121241.1121248>
- [8] S. Bashyal and G. K. Venayagamoorthy, "Human swarm interaction for radiation source search and localization," in *Swarm Intelligence Symposium (SIS)*. IEEE, 2008, pp. 1–8.
- [9] A. Kolling, S. Nunnally, and M. Lewis, "Towards human control of robot swarms," in *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*. ACM, 2012, pp. 89–96.
- [10] J.-P. de la Croix and M. Egerstedt, "Controllability characterizations of leader-based swarm interactions," in *2012 AAAI Fall Symposium Series*, 2012.
- [11] B. Donald, C. Levey, C. McGray, I. Paprotny, and D. Rus, "An untethered, electrostatic, globally controllable MEMS micro-robot," *J. of MEMS*, vol. 15, no. 1, pp. 1–15, Feb. 2006.
- [12] B. Donald, C. Levey, and I. Paprotny, "Planar microassembly by parallel actuation of MEMS microrobots," *J. of MEMS*, vol. 17, no. 4, pp. 789–808, Aug. 2008.
- [13] S. Floyd, E. Diller, C. Pawashe, and M. Sitti, "Control methodologies for a heterogeneous group of untethered magnetic micro-robots," *Int. J. Robot. Res.*, vol. 30, no. 13, pp. 1553–1565, Nov. 2011.
- [14] E. Diller, J. Giltinan, and M. Sitti, "Independent control of multiple magnetic microrobots in three dimensions," *The International Journal of Robotics Research*, vol. 32, no. 5, pp. 614–631, 2013. [Online]. Available: <http://ijr.sagepub.com/content/32/5/614.abstract>
- [15] D. Frutiger, B. Kratochvil, K. Vollmers, and B. J. Nelson, "Magmites - wireless resonant magnetic microrobots," in *IEEE Int. Conf. Rob. Aut.*, Pasadena, CA, May 2008.
- [16] S. Tottori, L. Zhang, F. Qiu, K. Krawczyk, A. Franco-Obregón, and B. J. Nelson, "Magnetic helical micromachines: Fabrication, controlled swimming, and cargo transport," *Advanced Materials*, vol. 24, no. 811, 2012.
- [17] A. Becker and T. Bretl, "Approximate steering of a unicycle under bounded model perturbation using ensemble control," *IEEE Trans. Robot.*, vol. 28, no. 3, pp. 580–591, Jun. 2012.
- [18] A. Becker, C. Onyuksel, and T. Bretl, "Feedback control of many differential-drive robots with uniform control inputs," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2012.
- [19] K. Takahashi, K. Hashimoto, N. Ogawa, and H. Oku, "Organized motion control of a lot of microorganisms using visual feedback," in *IEEE Int. Conf. Rob. Aut.*, May 2006, pp. 1408–1413. [Online]. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1641906&isnumber=34383>
- [20] A. Becker, C. Onyuksel, T. Bretl, and J. McLurkin, "Controlling many differential-drive robots with uniform control inputs," *The international journal of Robotics Research*, vol. 33, no. 13, pp. 1626–1644, 2014.
- [21] A. Becker, G. Habibi, J. Werfel, M. Rubenstein, and J. McLurkin, "Massive uniform manipulation," in *IEEE International Conference on Intelligent Robots and Systems*, Nov. 2013.
- [22] J. Cazes, *Encyclopedia of Chromatography*, 2nd ed. New York: Taylor & Francis Group, 2005, vol. 2.