Stochastic Swarm Control with Global Inputs

Shiva Shahrokhi and Aaron T. Becker

***Abstract*— Micro and nano robots can be built in large numbers, but generating independent control inputs for each robot is prohibitively difficult. Instead, micro and nano robots are often controlled by a global field. In previous work we conducted large-scale human-user experiments where humans played games that steered large swarms of simple robots to complete tasks such as manipulate blocks. One surprising result was that humans completed a block-pushing task *faster* when provided with only the mean and variance of the robot swarm than with full-state feedback. This paper investigates controllers that use only the mean and variance of a robot swarm. We prove that the mean position is controllable, then provide conditions under which variance is controllable. We then derive automatic controllers for these and a hysteresis-based switching control for the first two moments of the robot distribution. Finally, we employ these controllers as primitives for a block-pushing task.**

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.

II. Related Work

A. www.SwarmControl.net

The goal of [4] was 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.

S. Shahrokhi and A. Becker are with the Department of Electrical and Computer Engineering, University of Houston, Houston, TX 77204-4005 USA {sshahrokhi2, atbecker}@uh.edu

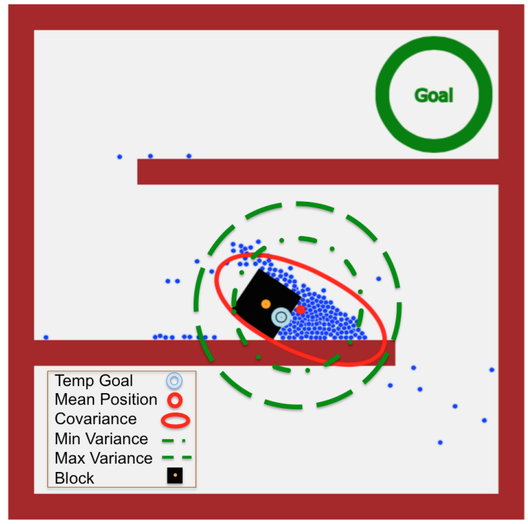

Fig. 1. A swarm of robots, all controlled by a uniform force field, can be effectively controlled by a hybrid controller that knows only the first and second moments of the robot distribution. Here a swarm of simple robots (blue discs) pushes a green block toward the goal.

Our previous work with over a hundred hardware robots and thousands of simulated robots [5] 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. To gather better data, we designed SwarmControl.net, a large-scale online game to test how humans interact with large swarms. Our goal was to test several scenarios involving large-scale human-swarm interaction (HSI), and to do so with a statistically-significant sample size. These experiments [4] showed that numerous simple robots responding to global control inputs are directly controllable by a human operator without special training, and that the visual feedback of the swarm state should be very simple in order to increase task performance. All code [6], and experimental results are posted online.

B. Block pushing and Compliant Manipulation

```
copy my block pushing references, look
up compliant manipulation
```

C. Human-Swarm Interaction

Olson and Wood studied human *fanout*, the number of robots a single human user could control [7]. 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 [8].

A number of user studies compare methods for controlling large swarms of simulated robots, for example [9]–[11]. 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 [11] was limited to a pool of 18 participants, [9] 5, and [10] 32. Using an online testing environment, we conduct similar studies but with much larger sample sizes.

D. 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 [12], [13]; magnetic structures with different cross-sections that could be independently steered [14], [15]; *MagMite* microrobots with different resonant frequencies and a global magnetic field [16]; and magnetically controlled nanoscale helical screws constructed to stop movement at different cutoff frequencies of a global magnetic field [1], [17].

Similarly, our previous work [18], [19] 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.

E. 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], [12]–[17], [20] 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 [19] and at worst a matrix inversion [18]. 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

Consider holonomic robots that move in the 2D plane. We want to control position and velocity of the robots. First, assume a noiseless system and with just one robot. Our inputs are global forces $[u_x, u_y]$.

$$\begin{bmatrix} \dot{p}_x & = & v_x \\ \dot{v}_x & = & \frac{1}{m} u_x \end{bmatrix}, \begin{bmatrix} \dot{p}_y & = & v_y \\ \dot{v}_y & = & \frac{1}{m} u_y \end{bmatrix} \tag{1}$$

The state-space representation in standard form is:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \tag{2}$$

We define our state vector as:

$$[x_1, x_2, x_3, x_4]^\intercal = [p_x, v_x, p_y, v_y]^\intercal,$$

and our state space representation as:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{1}{m} & 0 \\ 0 & 0 \\ 0 & \frac{1}{m} \end{bmatrix} u \tag{3}$$

We want to find number of states that we can control, which is given by the rank of the *controllability matrix*

$$\mathcal{C} = \{B, AB, A^2B, ..., A^{n-1}B\}. \tag{4}$$

$$\text{Here } \mathcal{C} = \left\{ \begin{bmatrix} 0 & 0 \\ \frac{1}{m} & 0 \\ 0 & 0 \\ 0 & \frac{1}{m} \end{bmatrix}, \begin{bmatrix} \frac{1}{m} & 0 \\ 0 & 0 \\ 0 & \frac{1}{m} \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \ldots \right\} \tag{5}$$

And thus all four states are controllable

B. Independent control with multiple robots is impossible

A single robot is fully controllable, but what happens with n robots? For holonomic robots, movement in the x and y coordinates are independent, so for notational convenience without loss of generality we will focus only on movement in the x axis. Given n robots to be controlled in the x axis, there are $2n$ states, n positions and n velocities.

$$[x_1, x_2, \dots, x_{2n-1}, x_{2n}]^\intercal = [p_{x,1}, v_{x,1}, \dots, p_{x,n}, v_{x,n}]^\intercal$$

Our state-space representation is:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_{2n-1} \\ \dot{x}_{2n} \end{bmatrix} = \begin{bmatrix} 0 & 1 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & \dots & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{2n-1} \\ x_{2n} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u_x \quad (6)$$

However, just as with one robot, we can only control two states because $\mathcal{C}$ has rank two:

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

C. Controlling Mean Position

This means any number of robots controlled by a global command, have just two controllable states in each axis. We can not control position of all the robots, but what states are controllable? To answer this question we create the following reduced order system that represents the average position and velocity of the n robots:

$$\begin{bmatrix} \dot{\bar{x}}_p \\ \dot{\bar{x}}_v \end{bmatrix} = \frac{1}{n} \begin{bmatrix} 0 & 1 & 0 & 1 & \dots & 0 & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{2n-1} \\ x_{2n} \end{bmatrix} + \frac{1}{n} \begin{bmatrix} 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & 1 & \dots & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u_x \quad (8)$$

Thus:

$$\begin{bmatrix} \dot{\bar{x}}_p \\ \dot{\bar{x}}_v \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \bar{x}_p \\ \bar{x}_v \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \quad (9)$$

We again analyze $\mathcal{C}$:

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

This matrix has rank two, and thus the controllable states of the swarm are the average position and average velocity. ■ Due to symmetry, only the mean position and mean velocity are controllable. However, there are several techniques for breaking symmetry, for example by allowing independent noise sources [21], or by using obstacles [5].

D. Controlling the variance of many robots

The variance, σ^2, of the robot's position is computed:

$$\overline{x}(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} x_i, \qquad \sigma^2(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2 \quad (11)$$

Controlling the variance requires being able to increase and decrease the variance. We will list a sufficient condition for each. Both conditions are readily found at the micro and nanoscale. Real systems, especially at the micro scale, are affected by unmodelled dynamics. These unmodelled dynamics are dominated by Brownian noise. To model this (2) must be modified as follows:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + W\varepsilon(t) \quad (12) \\ y(t) &= Cx(t) + Du(t) \end{aligned}$$

where $\varepsilon(t)$ is the error in the system. Given a large free workspace, *Brownian noise* increases the variance linearly with time.

$$\dot{\sigma}^2(\mathbf{x}(t), u(t) = 0) = W\varepsilon$$

A flat obstacle can be used to decrease variance. Pushing a group of dispersed robots against a flat obstacle will decrease their variance until the maximum packing density is reached.

We will prove the origin is globally asymptotically stabilizable by using a control-Lyapunov function [22]. A suitable Lyapunov function is squared variance error:

$$V(t, \mathbf{x}) = \frac{1}{2}(\sigma^2(\mathbf{x}) - \sigma^2_{goal})^2 \quad (13)$$

$$\dot{V}(t, \mathbf{x}) = (\sigma^2(\mathbf{x}) - \sigma^2_{goal})\dot{\sigma}^2(\mathbf{x}) \quad (14)$$

We note here that $V(t, \mathbf{x})$ is positive definite and radially unbounded, and $V(t, \mathbf{x}) \equiv 0$ only at $\sigma^2(\mathbf{x}) = \sigma^2_{goal}$. To make $\dot{V}(t, \mathbf{x})$ negative semi-definite, we choose

$$u(t) = \begin{cases} \text{move toward wall} & \text{if } \sigma^2(\mathbf{x}) > \sigma^2_{goal} \\ 0 & \text{if } \sigma^2(\mathbf{x}) \leq \sigma^2_{goal}. \end{cases} \quad (15)$$

For such a $u(t)$,

$$\dot{\sigma}^2(\mathbf{x}) = \begin{cases} \text{negative} & \text{if } \sigma^2(\mathbf{x}) > \max(\sigma^2_{goal}, \sigma^2_{min}(n, r)) \\ W & \text{if } \sigma^2(\mathbf{x}) \leq \sigma^2_{goal}, \end{cases} \quad (16)$$

and thus $\dot{V}(t, \mathbf{x})$ is negative definite.

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

The mean and variance of the swarm cannot be controlled simultaneously, however if the dispersion due to Brownian motion is much less than the maximum controlled speed, we can adopt a hysteresis-based controller to regulate the mean and variance. Such a controller normally controls the mean position according to (18), but switches to controlling variance if the variance exceeds some σ^2_{max}. The variance is lowered to less than σ^2_{min}, and the system returns to controlling the mean position.

A key challenge is to select proper values for σ^2_{min} and σ^2_{max}. For large n, Graham and Sloan showed that

Algorithm 1 Mean and variance control

Require: Knowledge of swarm mean $[\bar{x}, \bar{y}]$, and $x_{min}, x_{max}, y_{min}, y_{max}$, the locations of the rat angular boundary.

```
loop
    Compute the ?
    Compute the ?
end loop
```

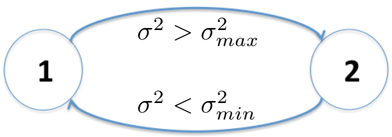

Fig. 2. Two states for controlling variances.

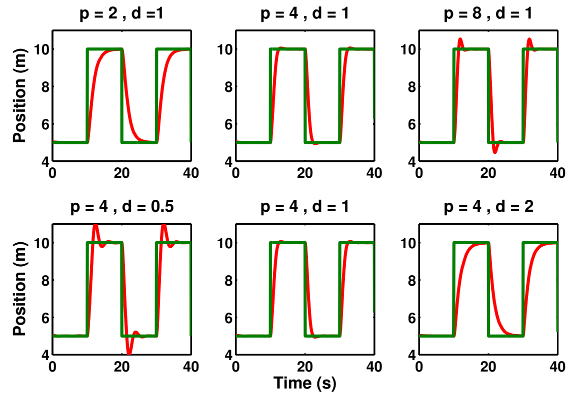

Fig. 3. Tuning proportional (K_p, top) and derivative (K_d, bottom) gain values improves performance. These plots show convergence with 100 robots.

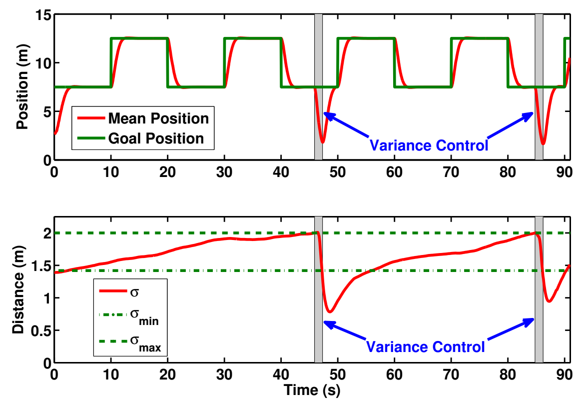

Fig. 4. Simulation result with 100 robots under hybrid control law (17), which controls both the mean position (top) and variance (bottom). For ease of analysis, only x position and variance are shown.

the minimum variance packing for n circles with radius r is $\approx 4r^2(\sqrt{3}n^2)/(4\pi) = 0.138n^2r^2$ [23]. The random packings generating by pushing our robots into corners are suboptimal, so we choose the conservative values $\sigma^2_{min} = (\frac{nr}{2})^2, \sigma^2_{min} = (\frac{nr3}{2})^2$.

$$switch = \begin{cases} 1 \rightarrow 2 & \sigma^2 > \sigma^2_{max} \\ 2 \rightarrow 1 & \sigma^2 < \sigma^2_{min} \end{cases} \tag{17}$$

This is a standard technique for dealing with control objectives that evolve at different rates [24], [25], and the hysteresis avoids rapid switching between control modes.

IV. Simulation

A. Controlling the mean position

For controlling mean position, we use a PD controller that uses the mean position and mean velocity. Our control input is the force applied to each robot, and goal positions are the desired positions:

$$\begin{aligned} u_x &= K_p(x_{goal} - \bar{x}) + K_d(0 - \bar{v}_x) \\ u_y &= K_p(y_{goal} - \bar{y}) + K_d(0 - \bar{v}_y) \end{aligned} \tag{18}$$

here K_p is the proportional gain, and K_d is the derivative gain. We performed a parameter sweep to identify the best values. Representative experiments are shown in Fig. 3. 100 robots were used and the maximum speed was 3 meter per second. We can achieve an overshoot of ?? and a rise time of ?? with $K_p = 4$, and $K_d = 1$.

(we can't say the best gain values are 4 and 1', but we can say 'to achieve these goals, we can use the gain values 4 and 1'.)

B. Controlling the variance

For variance control we use a hysteresis-based controller discussed in Section III-D. Waiting is sufficient to increase variance because Brownian noise naturally disperses the swarm in such a way that the variance increases linearly [26]. If faster dispersion is needed, the swarm can be pushed through obstacles such as a diffraction grating or Pachinko board [5]. To decrease the variance, we push the swarm into corners to decrease the variance. Our algorithm identifies the nearest corner by

how does it work?

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. Hybrid Control of mean and variance

Figure 4 shows representative results of the hybrid controller given in Alg. 1 with 100 robots in a square workspace with no internal obstacles.

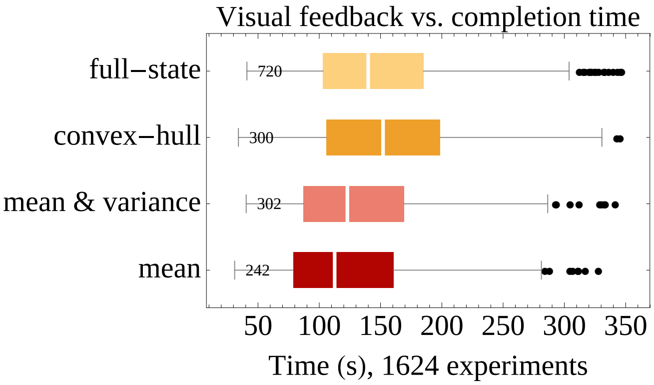

Fig. 5. Completion-time results for the four levels of visual feedback shown in Fig. 6. Surprisingly, players perform better with limited feedback–subjects with only the mean + variance outperformed all others.

V. Block-Pushing 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. The original experiment explored 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. 6 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 indicated the opposite: players with just the mean completed the task faster than those with full-state feedback. As Fig. 5 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 phenomenons that confuse predators—while working with just the mean + variance is like using a "spongy" manipulator. Our beta-testers found convex-hull feedback confusing and irritating. A single robot left behind an obstacle will stretch the entire hull, obscuring the majority of the swarm.

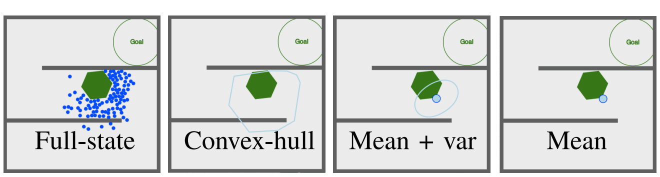

Fig. 6. Screenshots from task *Vary Visualization*. This experiment challenges players to quickly steer 100 robots (blue discs) to push an object (green hexagon) into a goal region. We record the completion time and other statistics.

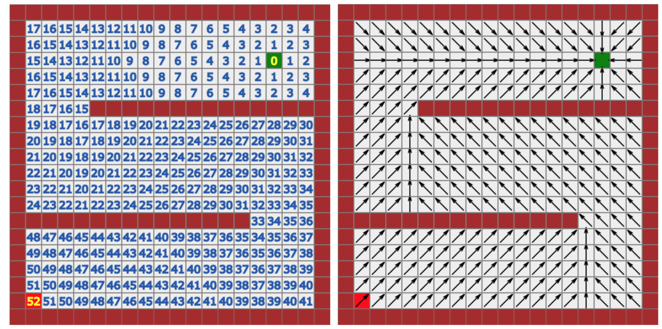

Fig. 7. The BFS algorithm finds the shortest path for the moveable block (left), which is used to compute gradient vectors (right).

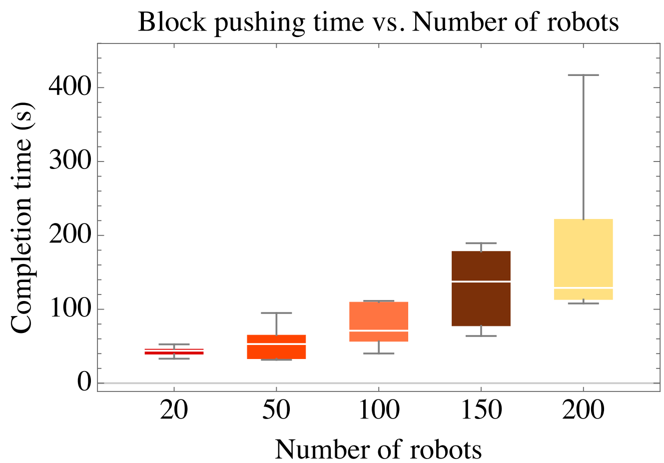

Fig. 8. Completion-time results using the automatic controller from Alg. 2 for different numbers of robots. Each bar represents the results of five experiments.

A. Automated Block Pushing

To solve this problem, we discretized the environment. On this discretized grid we used breadth-first search to determine M, the shortest path from any point for the block to the goal, and generated a gradient map $\nabla\mathbf{M}$ toward the goal as shown in Fig. 7. The blocks's center of mass is at $\mathbf{b}$ and has radius r_b. The robots were then directed to assemble at $\mathbf{b} - r_b\nabla\mathbf{M}$ to push the block toward the goal location.

Algorithm 2 Block-pushing controller for a robotic swarm.

Require: Knowledge of swarm mean $[\bar{x}, \bar{y}]$, the moveable block's center of mass $\mathbf{b}$, and the locations of all convex corners $\mathbf{C}$

loop
  Compute the ?
  Compute the ?
end loop

plot of results, comparing to humans

images of worlds where the algorithm fails, short discussion.

VI. Conclusion and Future Work

Inspired by large-scale human experiments with swarms of robots under global control, this paper investigated controllers that use only the mean and variance of a robot swarm. We proved that the mean position is controllable, and provided conditions under which variance is controllable. We derived automatic controllers for each and a hysteresis-based switching control that controls the mean and variance of a robot swarm. Finally, we employed these controllers as primitives for a block-pushing task. Future work should implement these controllers on a robot swarm, and decrease completion time by avoiding counter-productive contact with the block while the swarm is lowering its variance.

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.

[4] A. Becker, C. Ertel, and J. McLurkin, "Crowdsourcing swarm manipulation experiments: A massive online user study with large swarms of simple robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014. [Online]. Available: https://arxiv.org/submit/913241/view

[5] 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.

[6] C. Ertel and A. Becker. (2013, Sep.) Swarmcontrol git repository. [Online]. Available: https://github.com/crertel/swarmmanipulate.git

[7] 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.

[8] 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

[9] 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.

[10] 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.

[11] J.-P. de la Croix and M. Egerstedt, "Controllability characterizations of leader-based swarm interactions," in *2012 AAAI Fall Symposium Series*, 2012.

[12] 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.

[13] 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.

[14] 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.

[15] 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

[16] 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.

[17] 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.

[18] 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.

[19] 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.

[20] 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

[21] 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.

[22] A. M. Lyapunov, translated and edited by A.T. Fuller, *The General Problem of the Stability of Motion*. London: Tayor & Francis, 1992.

[23] R. L. Graham and N. J. Sloane, "Penny-packing and two-dimensional codes," *Discrete & Computational Geometry*, vol. 5, no. 1, pp. 1–11, 1990.

[24] S. Sadraddini and C. Belta, "Swarm manipulation with a uniform magnetic field," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[25] M. Kloetzer and C. Belta, "Temporal logic planning and control of robotic swarms by hierarchical abstractions," *Robotics, IEEE Transactions on*, vol. 23, no. 2, pp. 320–330, 2007.

[26] A. Einstein, *Investigations on the Theory of the Brownian Movement*. Courier Corporation, 1956.