Reply to Reviewer Comments: “Algorithms For Shaping a Particle Swarm With a Shared Control Input Using Boundary Interaction”

We thank the reviewers and the handling editors for their constructive feedback on our paper. We have corrected and improved it based on their helpful comments. We feel our paper addresses all their comments, and hope the reviewers and editors will now find the paper suitable for publication.

Point-by-point responses to Reviewer 1:

This paper presents algorithms for controlling a swarm of robots to take a shape by using uniform global inputs and boundary friction forces. Shaping a swarm of robots is a significant problem for a widely applications. There exist many research results about it.

R1-1: *In the section VI, this paper writes that “The techniques in this paper are inspired by fluid-flow techniques and artificial force- fields”. However, in section II, how to understand (10) and figure 4(c).*

R1-1: We grounded equation (1) with three physical examples at different size scales of swarms. We updated (10) with a standard linear model of boundary-layer fluid flow (adding reference [27]), and an explanation of Fig. 4c.

R1-2: *The paper is organized well. However, reviewer still suggests that the section VI should be moved to the front of the paper and…*

R1-2: We agree, and moved Section VI to the front to relate our swarm shape control techniques to the appropriate robotics disciplines.

R1-3: *…the main contributions of the paper should be described more clearly in the introduction. The multimedia video shows abundant simulation and experiment results.*

R1-3: We rewrote the introduction to better explain this paper’s contributions. In particular the first paragraph now explains three systems that could benefit from the techniques introduced.

Point-by-point responses to Reviewer-3

R3-1: *The paper deals with an interesting topic of controlling the shape of a swarm of weak controllable robots. The robots can only response to uniform global inputs and have infinite friction forces with boundary walls. Two kinds of shaping are introduced, one is to move individuals to certain locations and the other is to obtain certain variances as a swarm. The simulations of shaping robots to certain locations look interesting. In robotic applications, weak capability is often accompanied by huge quantity. As the population increasing, the total moves are almost exponentially increasing. It is probably unbearable to many* *potential applications, such as nano-robots to cure human disease. My impression of such robots is the huge number of members. Also, the friction between the robot and boundary wall is assumed to be infinite. In real world, a robot can hardly keep his position with a large force paralleling with the wall. Although the situation can be alleviated by taking actions, it is not clear whether it will be uncontrollable when total moves increase.*

R3_2: Excellent discussion. We now highlight two physical systems (Fig. 1 and Fig. 6) that do have infinite friction parallel to the boundaries. Infinite friction is a limit case. Our other algorithms (II-A and the new correlation control in IV.C) do not require infinite friction.

R3-2 *Although as a type of shaping, it seems to me that adjusting the variances of the swarm is isolated with changing locations. The controlling in III.(c) is expected to be related to II.(a). I would appreciate more analysis details about their connection, for example, why the swarm should pushed into left or bottom when the \sigma_*^2 is larger or smaller than expected, how and how much it should move, instead of "swarm pushed into the ** wall untill **". In addition, it is better to describe more about the scenarios of variance controlling.*

R3_2: You are correct. For clarity of explanation, we have replaced IV.C with analytical results of a swarm being reshaped using boundary layer friction (10), and calculated the maximum correlation possible. We added a comparison of controlling correlation using boundary frictional forces with the “Using boundaries for stable configurations” in Sec III. Controlling the mean and variance of a swarm is the focus of our paper [10].

R3-3: *I guess a minor issue is presented in Eqn.(4), x-\bar{x}.*

R3-3: Thank you for catching this – we have fixed Eq. 4.

Point-by-point responses to Reviewer-4

The manuscript presents algorithms that form a particle swarm in a desired shape using a shared control input in a closed workspace. For a particle swarm to aggregate in a desired form, the manuscript first introduces a framework with constant force acting on the swarm and then applies wall friction to the swarm. The algorithms are validated in simulations and experiments.
The ideas of swarming a group of robots, presented in the manuscript, sound interesting, but the clarity of their presentation in the manuscript is not very good. In fact, the way of presenting ideas often raises questions. I recommend the rejection of the manuscript. Detailed comments follow.

R4-1: *Symbols x,y,u_x,u_y in Equation (1) are not introduced at all.*

R4-1: Thank you. We added explanation in paragraph 1 and a justification for this model.

R4-2: *In Pg. 1, Sect. I, line 8-9: Are (1) and (2) items in a list or Do they refer to Equations (1) and (2)?*

R4-2: These were items in a list. We deleted them to avoid confusion.

R4-3: *The Introduction section does not well explain the problem the authors are dealing with and does not even review any relevant existing work from which we can guess the main ideas of the manuscript.*

R4-3: Thank you for your feedback. As per your suggestion, we rewrote the introduction and moved the related work from the end to Section II.

R4-4: *In Pg. 1, Sect. II.A, line 4-6: The mean, covariance and correlation of a swarm in the x and y components are used here. Since x and y are not introduced when Equation (1) is introduced, it is unclear what they are. Are x and y probabilities of something?*

R4-4: We have added an explanation that x and y represent the position.

R4-5: *In Pg 2, line 7-8: What is the range of A? From Figs. 2 and 3, I could find out A ranges from 0 to 1, but it was not clear in the main context.*

R4-5: Excellent catch. We added range to the description of A: from 0 to 1.

R4-6: *How theories in Sect. II are applied in Sect. III is not clear. For example, which part in Alg. 1 or 2 relates to wall friction?*

R4-6: Helpful feedback from reviewers led to a re-work of the two-particle position control. Alg 1 replaces old Alg 1 and 2. Alg. 1 exploits the position dependent friction model.

R4-7: *How does DriftMove() work? It is not explained.*

R4-7: Thank you. The n robot position control algorithm relies on a **DriftMove(α,β,ε)** control input, shown in Fig. 7. Sec. IV.B, paragraph 3 now states: “A drift move consists of repeating a triangular movement sequence $\{(\beta/2, -\varepsilon), (\beta/2, \varepsilon), (-\alpha, 0)\}$. The robot touching a top wall moves right β units, while robots not touching the top move right $\beta - \alpha$.”

Point-by-point responses to Reviewer-5

This paper presents a shaping control algorithm for swarming of multiple robots that use global inputs in the closed area. The global inputs in this paper mean all the robots have the same velocity command in the 2-D plane.

The main point of the paper is to determine velocity command for polygonal shapes. Authors propose two algorithms: open-loop and closed-loop algorithms.

R5-1: *First of all, motivation of this research is weakened due to the type of global inputs. Motivated by small size robots lack of on-board computation, this work uses identical control input to all the robots.*

R5-1: Thank you. We rewrote the introduction to better discuss the motivation:

Particle swarms propelled by a global field are common in applied mathematics, biology, and computer graphics. (…) These system dynamics represent particle swarms in low-Reynolds number environments, where viscosity dominates inertial forces and so velocity is proportional to input force [1]. In this regime, the input force command u(t) controls the velocity of the robots. The same model can be generalized to particles moved by fluid flow where the vector direction of fluid flow u(t) controls the velocity of particles, or for a swarm of robots that move at a constant speed in a direction specified by a global input u(t) [2]. Our control problem is to design the control inputs u(t) to make all n particles achieve a task. As a current example, micro- and nano-robots can be manufactured in large numbers, see [3, 4, 5, 6, 7, 8, 9].

R5-2: *However, the identical control input of the paper is velocity command that requires on-board computation. For example, the reference [1] of previous work [2] in the paper needs the voltage of an actuator to drive a micro robot. The voltage command cannot be generated from velocity command without on-board computation. Using acceleration command is fit for your research.*

R5-2: The original paper was unclear. The introduction has been rewritten to address this misunderstanding. These system dynamics represent particle swarms in low-Reynolds number environments, where viscosity dominates inertial forces and so velocity is proportional to input forces. In this regime, the input force command **u** controls the velocity of the robots. The same model can be generalized to particles moved by fluid flow where the vector direction of fluid flow **u** controls the velocity of particles, or for robots that move at a constant speed in a direction specified by a global input **u**.

R5-3: *The proposed open-loop algorithm called Algorithm 2 has critical error. Since there is no restriction on choosing initial and ending position of robots, consider two robots in the 2-D plane; the first robot is located at (3,4), and the second robot is located at (2,4). Assume the ending position of the first robot is*

(5,4), the ending position of the second robot is (7,4), and the wall length is 7. At the line 8 in the algorithm, the first robot would be at (7,4), and the second robot at (6,4). And then algorithm keeps changing the y-axis position with epsilon value because of line 15, but x-axis positions of the robots are fixed. The algorithm has infinite loop with remaining that the x-axis positions of two robots never change because there is no stopping criteria. This shows that the algorithm does not work for all the potential cases.

R5-3: Thank you for this catch! Rewriting this algorithm was challenging because of the many corner cases, but your catch led to a more beautiful and concise algorithm that exploits rotational symmetry. The previous paper's Alg. 1 and 2 are replaced by the new Alg. 1. The video attachment shows the corner case you mentioned, and several others.

R5-4: *For closed-loop algorithm, global inputs keep generated until covariance and variance of the swarm reach desired certain values. However, the algorithm does not control the mean position of the swarm.*

R5-4: To better focus the paper, we removed the closed-loop controllers and replaced these with analytical results that match our hardware assumptions and hardware experiments. Our paper on mean control of a swarm is [10].

R5-5: *Although the paper says that mean position of swarm is moved to the center of the workspace, there is no evidence of the mean position at the center of the workspace. In addition, any control law and logic are not described for controlling the mean position of the swarm in the closed-loop algorithm. This paper insists that the algorithm is robust. However, several simulations and 100 robots experiments does not provide us with the robustness of your algorithm. Your algorithm depends on the size of closed area and the number of robots. If the closed area is packed with many robots, your algorithm may not be work*

R5-5: Quite right. We have rewritten the algorithms to make clear that all analytical results are functions of the workspace size and the size of the particle swarm.

R5-6: *The abstract of your paper says your algorithm enables the stable swarm. However, this paper does not show any stability and convergence proof.*

R5-6: "Stability" was ambiguous. In this paper, a stable configuration of a swarm under a constant global input is the minimum energy solution. To clarify this distinction, we changed the title of II.A. from "Using Boundaries: Fluid Settling In a Tank" to "A. Using Boundaries: stable configurations of a swarm". We also edited the text in this section.

R5-7: *Equations (5) and (6) are wrong. Equation (5) is 10*sqrt(2)/6*sqrt(-A*tan(beta)). Equation (6) is 10*sqrt(2)/6*sqrt(-A*cot(beta)). Since A is positive, mean position is imaginary, which means both equations are wrong.*

R5-7: Excellent catch. This was an unfortunate typo. This is changed to "the mean when R is a triangular region in the lower-left corner is:"

Equation 5 was changed to

$$\bar{x}(A,\beta) = \frac{\int_0^{\sqrt{2A\tan(\beta)}} \left(\int_0^{\cot(\beta)\left(\sqrt{2A\tan(\beta)}-x\right)} dy \right) x\,dx}{A} = \frac{\sqrt{2}}{3}\sqrt{A\tan(\beta)} \tag{5}$$

Equation 6 was changed to

$$\bar{y}(A,\beta) = \frac{\int_0^{\sqrt{2A\tan(\beta)}} \left(\int_0^{\cot(\beta)\left(\sqrt{2A\tan(\beta)}-x\right)} y\,dy \right) dx}{A} = \frac{\sqrt{2}}{3}\sqrt{A\cot(\beta)} \tag{6}$$

R5-8: *This paper does not follow IEEE format. IEEE Editorial system manual tells us that there is no symbol in front of section description. For example, § should not be used before section number.*

R5-8: Thank you for bringing this to our attention. The symbol was removed throughout.

R5-9: *This paper has a significant number of ambiguous expressions. What is gravity pointing? Your robot moves in the 2-D plane where no gravity affects the motion of the robot.*

R5-9: This was ambiguous. Our 2D plane workspace could be oriented in any direction, and cause an external force (such as gravity) to pull in the direction β relative to the workspace. The paper now says: "they move inside a square workspace under the influence of a force pulling in direction β".

R5-10: *What do you mean they flow like water?*

R5-10: This was also ambiguous. The paper now states "Under a global input, the swarm moves to a side of the workspace and forms a polygonal shape that minimizes potential energy, as shown in Fig. 2, see also [24]."

***R5-11:** What do mean Equation (9) is not position dependent friction? You already said friction is infinite, which means no dependency of position.*

R5-11: Friction is a function of distance from boundary walls. The infinite friction model is infinite only at the boundary. We added a description of eq. 10.

***R5-12:** What is region R?*

R5-12: We added an explanation of R: "The region of integration R is the polygon containing the swarm."

***R5-13:** What is a constant area A?*

R5-13: We added an explanation of A: "The swarm is large, but the robots are small in comparison, and together occupy a constant area A, A in [0, 1]."

***R5-14:**In addition, the style of reference is not consistent. In the introduction of the paper, [1],[2],[3],[4],[5],[6], or [7] are written. In the related work section, [1, 21, 22] are written. Please use one reference style.*

R5-14: Good point. We are now doing the condensed style.

***R5-15:**In terms of multimedia, video quality is good overall, but irrelevant video clips in your video weaken your contribution. The first clip shows kilobot robots programmed to move toward a bright light. This clip cannot tell your problem or support your algorithm based on global input because individual kilobot has its own micro controller that enables on-board computation. Further, reference [3] related to kilobot already achieved the various shape similar to your work.*

R5-15: There are several key distinctions. Our work in [3] used small populations swarms ranging in size from 1 to 3. Ref [26] showed several demonstrations with human controllers, while this paper presents analytical models on shape control, uses boundary friction, has numerous automated hardware experiments, and has relevance for future work. The position control technique in [26] relied on a restrictive obstacle model, while biological and engineered structures often have boundary layer friction that can be exploited.

References [15] and [16] used kilobots that communicated, had autonomy, used a map, and moved only a small fraction of the swarm at a time. This paper investigates large particles swarms (10^2—10^8) with no inter-robot communication, no autonomy, no maps (on the robots), and moves the entire swarm with a shared control input.

***R5-16:** The second clip shows your previous work. This does not show your current work. Other video clips well present your work.*

R5-16: Thank you. We added a caption to explain how the short clip near the beginning motivated this research showing that this short clip refers to our previous work in [26] with **human-steered** swarms. Our new work uses an automated approach.

Statement of Changes

A side-by-side diff of submission 1 and 2 is available at https://draftable.com/compare/UHSQurFaJTKsTojA

Changes:

1. Rewrite of Introduction, explaining eq (1) and relevant examples
2. Related work moved from paper end to be Section II
3. New title for III-A "Using Boundaries: stable configurations of a swarm"
4. Rewrite of III-A
5. New Friction Model, Fig. 4.C
6. New Alg 1 "GenerateDesiredSpacing" in Section IV, replaces old Alg. 1 and 2.
7. New section IV- C. "Maximizing Correlation Using Wall Friction", replaces heuristic-based controller with analytical model.
8. New section V-C and Figure 10.
9. Numerous small corrections to improve flow, correct grammar, and to reduce ambiguity.