**Summary**

This paper contains some interesting results on shape control of a robot swarm. However, the paper has major flaws. It is unorganized, containing list of various descriptions without a clear theme and sufficient justification of the method used and the results collected. Reviewers are questioning the technical correctness of the algorithms and the soundness of technical approaches due to the lack of clarity.

**Review 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. In the section VI, this paper writes that “The techniques in this paper are inspired by fluid-flow techniques and artificial force- fields” .

1.) However, in section II, how to understand (10) and figure 4(c)?

2.)The paper is organized well. However, reviewer still suggests that the section VI should be moved to the front of the paper and

3.) the main contributions of the paper should be described more clearly in the introduction. The multimedia video shows abundant simulation and experiment results.

**Review 3:**

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 paralling 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. 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. Shiva: I agree. We didn’t tell why we did the covariance control, and we didn’t fully describe it. I will go ahead and add some description.

I guess a minor issue is presented in Eqn.(4), x-\bar{x}. Shiva: I corrected that

**Review 4: (When you review a paper, do not recommend accept/reject in the review)**

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.

1)\* Symbols x,y,u\_x,u\_y in Equation (1) are not introduced at all. Shiva: I added a description, HOWEVER, we are wrong about u as a velocity. This is the next reviewer’s problem.

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)? Shiva: Deleted them

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. Shiva: SEREIOUSLY? OK! I will add some more explanation; maybe we have to add the related work here?

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? Shiva: I have added that x and y are positions, so it should be resolved.

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. Shiva: I added the range to the description of A.

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? \* How does DriftMove() work? It is not explained. Shiva: I think this part needs more explanation. Like: telling why we didn’t just use the corners, and then we used the wall friction. Also telling

**Review 5: (probably a grumpy, but a careful reader, PhD student)**

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. 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. Shiva: What? I guess we were not clear about our control input. Light just shows the direction, how do we have velocity controlled? I guess this misunderstanding has occurred because Eq. 1. Where we introduce velocity as our control input. Adding this may help: We used light to control the direction of the robots, and all the robots have almost a constant velocity to go toward the lights. We do not control the velocity.

**Better, in intro: These system dynamics represent particle swarms in low-Reynolds number environments, where viscosity dominates inertial forces and so velocity is proportional to input forces [cite Purcell, life at low Reynolds numbers]. In this regime, the input force command $\mathbf{u}$ 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 $\mathbf{u}$ controls the velocity of particles, or for robots that move at a constant speed [kilobots cite] in a direction specified by a global input $\mathbf{u}$.**

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. Shiva: We are using acceleration for the simulation, we are just controlling the direction and have a constant speed for the kilobots. We need to tell both.

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.

Shiva: First I thought he is wrong, and got irritated. But he is right. The algorithm doesn’t work in this critical positions, we have to modify it.

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. Shiva: We are only controlling mean actually. 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. TRUE. If the closed area is packed with many robots, your algorithm may not be work. TRUE. The abstract of your paper says your algorithm enables the stable swarm. However, this paper does not show any stability and convergence proof.

**Change title of II.A. “Using Boundaries: Fluid Settling In a Tank” to “A. Using Boundaries: stable configurations of a swarm”**

**Shiva: Done.**

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.

**Ouch – correct! Change to “the mean when $R$ is a triangular region in the lower-left corner is:”**

**And change the math for 5 to**

**\frac{\int\_0^{\sqrt{2 A \tan (\beta)}} \left(\int\_0^{\cot (\beta) \left(\sqrt{2 A \tan (\beta)}-x\right)} x \, dy\right) \, dx}{A}**

**And change the math for 6 to**

**\frac{\int\_0^{\sqrt{2 A \tan (\beta)}} \left(\int\_0^{\cot (\beta) \left(\sqrt{2 A \tan (\beta)}-x\right)} y \, dy\right) \, dx}{A}**

**Shiva: did that!**

This paper does not follow IEEE format. IEEE Editorial system manual tells us that there is no symbol in front of section description. Shiva: didn’t know about it. I changed all the sections with Sec. For example, § should not be used before section number. This paper has a significant number of ambiguous expressions.

1)What is gravity pointing?

2)Your robot moves in the 2-D plane where no gravity affects the motion of the robot.

3)What do you mean they flow like water?

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

5)What is region R? What is a constant area A? Shiva: I added explanation to them

6)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. Shiva: Changed the style to one.

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. Shiva: We are not clear here. Further, reference [3] related to kilobot already achieved the various shape similar to your work. Shiva: What is the benefit of this work over previous work?

The second clip shows your previous work. This does not show your current work. Shiva: Should I delete that part of our clip?

Other video clips well present your work. Shiva: Thank you for this only one complement at the end.