Collecting a Swarm in a 2D Environment Using Shared, Global Inputs

Shiva Shahrokhi and Aaron T. Becker

***Abstract*— Consider a swarm of robots initialized in a 2D planar grid world where each position is either free-space or obstacle. Assume all robots respond equally to the same global input, such as an applied fluidic flow or electric field. This paper investigates what input sequence should be applied to collect all the robots to one position.**

I. INTRODUCTION

For many therapeutic treatments it is important to concentrate a drug at a particular site. The old adage *toxicity is a function of concentration* explains that often we can flow a diluted drug through the body without ill-effect, and then kill cells at a targeted location by collecting drug particles. Targeted drug therapy is a goal for many interventions, including treating cancers, delivering pain-killers, and stopping internal bleeding.

This paper builds on the techniques for controlling many simple robots with uniform control inputs presented in [1]–[3], and outlines new research problems.

a) Definitions: The ‘robots’ in this paper are simple particles without autonomy. A planar grid *workspace* W is filled with a number of unit-square robots (each occupying one cell of the grid) and some fixed unit-square blocks. Each unit square in the workspace is either *free*, which a robot may occupy or *obstacle* which a robot may not occupy. Each square in the grid can be referenced by its Cartesian coordinates $\boldsymbol{x} = (x, y)$. All robots are commanded in unison: the valid commands are “Go Up” (u), “Go Right” (r), “Go Down” (d), or “Go Left” (l).

We consider two classes of commands, discrete and maximal moves. *Discrete moves*: robots all move in the commanded direction one unit unless they are prevented from moving by an obstacle or a stationary robot. *Maximal moves*: robots all move in the commanded direction until they hit an obstacle or a stationary robot. For maximal moves, we assume the area of W is finite and issue each command long enough for the robots to reach their maximum extent. A *command sequence* $\boldsymbol{m}$ consists of an ordered sequence of moves m_k, where each $m_k \in \{u, d, r, l\}$ A representative command sequence is $\langle u, r, d, l, d, r, u, \ldots \rangle$.

We consider two notions of a collected swarm: if robots are allowed to overlap, the swarm is connected when all robots share the same (x, y) coordinates. If the robots may not overlap, the swarm is collected when it forms one 4-connected component.

Research questions include

1) What connected topologies allow collecting a swarm?

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

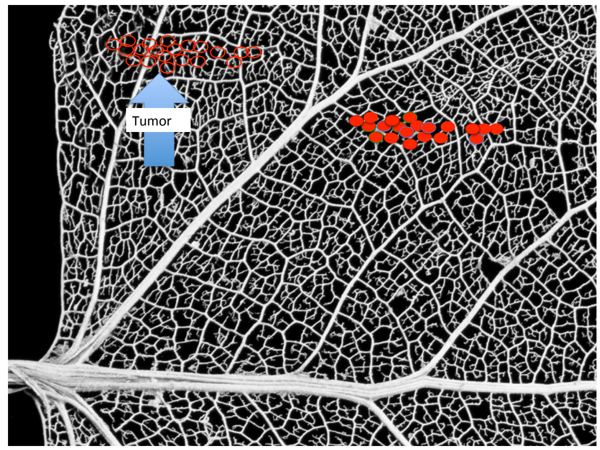

Fig. 1. Vascular networks are common in biology such as the circulatory system and cerebrospinal spaces, as well as in porous media including sponges and pumice stone. Navigating a swarm using global inputs, where each member receives the same control inputs, is challenging due to the many obstacles. This paper focuses on using boundary walls and wall friction to break the symmetry caused by the global input and control the shape of a swarm.

2) What connected topologies do not allow collecting a swarm?
3) What connected topologies can be collected with a given command sequence?

II. RELATED WORK

A. Localization

The strongest parallel in the literature is the field of *Almost sensor-less localization* or ”localizing a blind robot in a known map”, where a mobile robot using a compass, a bump-sensor that detects when the robot contacts a wall, and a map of the workspace must localize itself in the map [4], [5]. This has been recently extended to robots with bounded uncertainty in their inputs [6]. The basic methodology is to construct motion plans that: “First, actions are selected which reduce the uncertainty in the robots position to a finite set of possibilities. Second, actions are selected to reduce the uncertainty from this finite set to a single point. ” [4].

“The problem of finding a localizing sequence for a given environment can be seen as a planning problem in which the initial state is unknown and the current state is unobservable. To manage this uncertainty, we transform the problem from an unobservable planning problem in state space to an observable problem in a more complex space called the robots information space, which we now define. ” [4].

"it remains an open problem to generate localizing sequences that are optimal." [4].

This work assumes there is only one robot, and works in a simply connected polygonal environment. We want to relax this to handle unit robots.

Also related is work on sensorless part orientation, where flat tray is tilted in a series of directions to bring a polygonal part, initially placed at random orientation and position in the tray, to a known position and orientation [7]. This may be similar to localizing a robot with minimum travel, but this moves the robot and has it take additional measurements [8]

B. Robot Rendezvous

Rendezvous usually concerns two independent, intelligent agents that must meet at a predetermined time.

C. Manufacturing

This is similar to work on draining a polygon [9], but with discrete inputs.

Or placing ports for injection-molding

scholar search on this keyword

D. Robot Gathering

scholar search on this keyword

III. Our Problems of Interest

A. Topologies that do not allow collecting a swarm

Robots initialized in two unconnected components i and j of a freespace cannot be collected. The proof is trivial, since robot in freespace i can not reach freespace j.

Under maximal inputs, the freespace can be constructed with spaces resembling bottles or fish weirs from which a single robot cannot escape, as shown in Fig. [TODO]. If the freespace contains at least two such bottles with at least one robot in each, the swarm cannot be collected.

The world must be bounded. Two initially-separated robots in an unbounded world without obstacles cannot be collected, however with discrete inputs, one obstacle is sufficient [10].

B. connected topologies allow collecting a swarm

With discrete inputs and robots that are allowed to overlap, the problem can be reduced to localizing a sensorless robot in a known workspace. This is similar to work on draining a polygon [9], or localizing a blind robot [4], [5], but with discrete inputs.

Algorithm 1, brute-force-search: initializes a tree with the root node $\{C_0,\}$, where C_0 is the current configuration of possible robot locations, where C_0 is an ordered vector containing the locations of every possible location for a robot in the workspace and $\{\}$ is the shortest corresponding movement sequence to achieve this configuration. We then construct a breadth-first tree of possible configurations $\{u, r, d, l\}$, pruning leaves that already exist in the tree. We

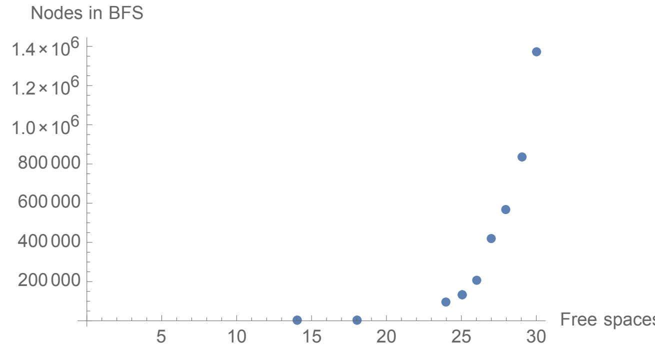

Fig. 3. Unfortunately, the optimal solution requires time that increases (approximately) exponentially with the number of free spaces.

stop with the cardinality at a leaf $C_i = 1$, which indicates that the swarm has been collected (equivalently, the robot has been localized). This algorithm produces the optimal path, but requires O(XX) time to learn and O(YY) memory.

Algorithm 1 BlindLocalizationBruteForce(W, C_0

Require:

Ensure: $C_{\boldsymbol{m}} \leftarrow \text{ApplyMoves}[\boldsymbol{m}, C_0, W]$, $|C_{\boldsymbol{m}}| = 1$

1: $T \leftarrow C_0$,

2: **return** (r_1, r_2)

Algorithm 2, greedy-approach:

1) possible configurations = current config
2) initial cardinality = current config
3) construct a breadth-first tree of possible configurations $\{u, r, d, l\}$ prune leaves that already exist.
4) stop when the cardinality of possible positions in any leaf is less than initial cardinality, apply this move, and restart the algorithm until cardinality is 1

Given two robots in a bounded $L \times L$ workspace, the worst case to bring them to the same position is $< L^2$. For n robots, $O(nL^2)$.

IV. Connected Topologies That Allow Collecting a Swarm

Assume that passageways are large compared to the maximum diameter of a swarm. We can accomplish by discritizing the environment, and solving the problem with discrete inputs.

References

[1] A. Becker, E. Demaine, S. Fekete, G. Habibi, and J. McLurkin, "Reconfiguring massive particle swarms with limited, global control," in *International Symposium on Algorithms and Experiments for Sensor Systems, Wireless Networks and Distributed Robotics (ALGOSENSORS)*, Sophia Antipolis, France, Sep. 2013, pp. 51–66.

[2] A. Becker, E. Demaine, S. Fekete, and J. McLurkin, "Particle computation: Designing worlds to control robot swarms with only global signals," in *IEEE International Conference on Robotics and Automation (ICRA)*. Hong Kong: IEEE, May 2014, pp. 6751–6756.

[3] A. Becker, E. D. Demaine, S. P. Fekete, G. Habibi, and J. McLurkin, "Reconfiguring massive particle swarms with limited, global control," in *Algorithms for Sensor Systems*, ser. Lecture Notes in Computer Science, P. Flocchini, J. Gao, E. Kranakis, and F. Meyer auf der Heide, Eds. Springer Berlin Heidelberg, 2014, vol. 8243, pp. 51–66. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-45346-5_5

[4] J. O'Kane and S. M. LaValle, "Almost-sensorless localization," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*. IEEE, 2005, pp. 3764–3769.

[5] J. O'Kane, "Almost-sensorless localization," Ph.D. dissertation, University of Illinois at Urbana-Champaign, 2005.

[6] J. S. Lewis and J. M. O'Kane, "Planning for provably reliable navigation using an unreliable, nearly sensorless robot," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1342–1357, 2013. [Online]. Available: http://ijr.sagepub.com/content/32/11/1342.abstract

[7] S. Akella and M. T. Mason, "Orienting toleranced polygonal parts," *The International Journal of Robotics Research*, vol. 19, no. 12, pp. 1147–1170, Dec. 2000.

[8] G. Dudek, K. Romanik, and S. Whitesides, "Localizing a robot with minimum travel," *SIAM Journal on Computing*, vol. 27, no. 2, pp. 583–604, 1998.

[9] G. Aloupis, J. Cardinal, S. Collette, F. Hurtado, S. Langerman, and J. ORourke, "Draining a polygon—or—rolling a ball out of a polygon," *Computational geometry*, vol. 47, no. 2, pp. 316–328, 2014.

[10] 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)*, Tokyo, Japan, Nov. 2013, pp. 520–527.

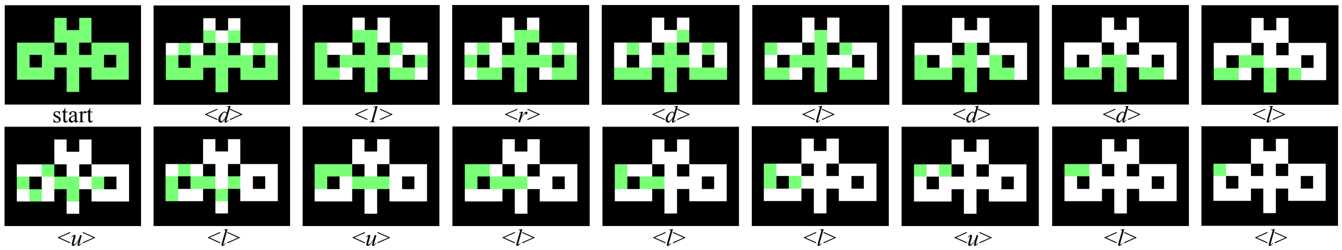

Fig. 2. With discrete inputs and robots that are allowed to overlap, the problem can be reduced to localizing a sensorless robot in a known workspace. Above shows the optimal solution for a map with 27 free spaces, which required expanding 423,440 nodes with an optimal path (shown) taking 17 moves.

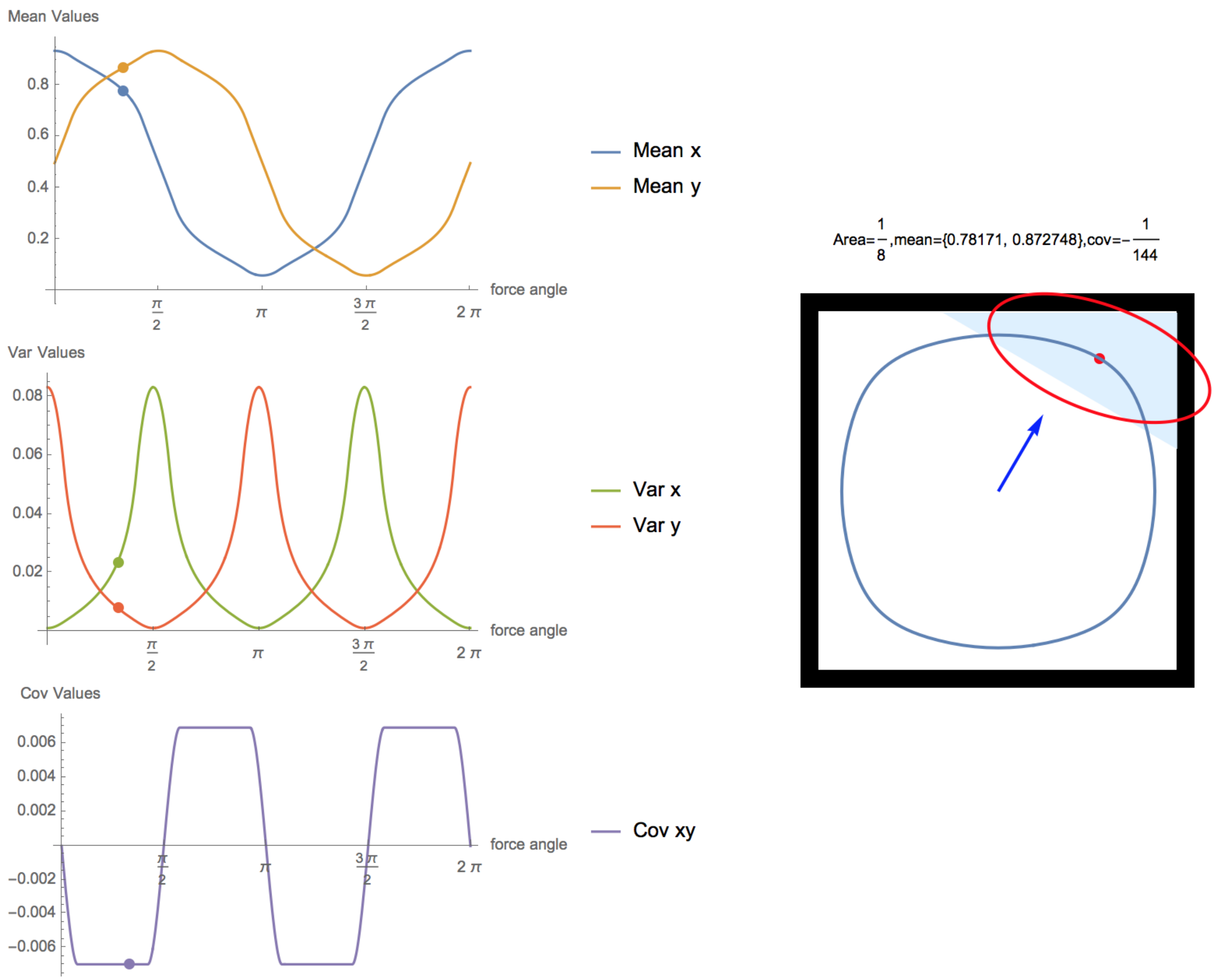

Fig. 4. a large-enough swarm controlled by global inputs can be modeled as a granular media (or as a fluid without surface tension). Above shows such a swarm in a square container. We assume at steady state under a global control command in the form of a vector direction, that the swarm reaches a minimum energy state with the highest energy level perpendicular to the applied force.