Open-Loop Collective Assembly Using a Light Field to Power and Control a Phototaxic Mini-Robot Swarm

Adam Bignell, Lily Li and Richard Vaughan[1]

***Abstract*— We propose a novel scheme that jointly addresses the problems of powering and coordinating a population of mini-robots for collective construction. In our setting, a population of simple mobile robots must push blocks into desired polygonal shapes. Each robot performs only simple phototaxis. Coordination is purely open-loop: a global light field guides and powers the robots. We demonstrate this concept in simulation and explore a series of dynamic light field design strategies that robustly result in assembled shapes including nonconvex polygons.**

I. INTRODUCTION

Collective assembly is a canonical multi-robot task whereby the robots construct pre-defined structures from either atomic building blocks [1] or the robot bodies themselves [2]. This topic has practical and theoretical motivations: we hope to speed construction with multiple robots working in parallel, and we seek to understand how to achieve globally coherent behavior from distributed robot systems under various constraints.

There are three basic approaches to coordinating multiple robots: (a) explicit planning in a global joint action space; (b) explicit local coordination by robot-robot communication; and (c) emergent coordination whereby the behavior of each robot may affect others without explicit communication. In the latter, robots can affect each other through, for example, physical interference or by modifying the environment. Here we consider emergent coordination due to its potential to scale to very large populations of cheap and simple robots. Our long term goal is to achieve massively scalable mini- or micro-scale construction, and we aim to avoid by design some of the disadvantages of other approaches.

In previous work we showed simple and very robust real-world box clustering performed by a swarm of 36 iRobot Creates [3]. The robots wander in a room containing the boxes they came in, initially scattered at random. The system shows the reliable emergence of tightly-packed and aligned box clusters over time, yet each Create executes only a semi-random walk designed for the Roomba robot for floor cleaning, and senses only collisions and a single nearby obstacle to one side.

Here we approach the more challenging task of constructing pre-defined polygonal shapes at specific locations, while seeking to keep the robots as simple as we can. In particular, we seek to avoid the need for robot localization or mutual detection. In our design, each robot uses only four light intensity brightness sensors, and computes only two integer subtractions per update.

[1]Autonomy Lab, Simon Fraser University. abignell@sfu.ca

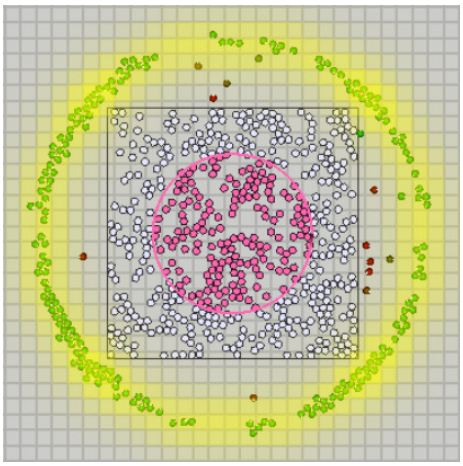


Fig. 1. Problem setting: Robots must push boxes to densely pack a target shape. Robots are circles, colored green if fully charged, red if discharged. Boxes are hexagons, colored pink if they are contained by the goal shape (pink circle), otherwise white. A programmable light field is implemented as an array of sources, colored gray to yellow by brightness. Pink is for visualization only: robots perceive only light brightness. The black square is a robot-permeable but box-impermeable wall.

All robot systems must solve the problem of supplying sufficient power for the duration of the task, and doing this for large numbers of small robots is very challenging. An iRobot Create operates for around 1-2 hours on a fully charged battery. The Kilobot, another robot frequently used in swarm construction, lasts for between 3 and 24 hours depending on activity [4]. Yet collective construction methods often take longer than this to complete. Emergent methods in particular usually achieve their effects by lots of energetically costly wandering and bumping. In conventional designs, additional charging facilities and behaviors are needed, implying added complexity and non-work overhead. In practice we see charging often abstracted away, ignored in simulation and performed manually by experimenters in real robots.

One solution to this problem is a powered floor, so robots are almost constantly powered. If the robots are small and simple enough to require little power, we can also use photocells and bright ambient light to power the robots without a special floor. If the robots are capable of phototaxis, light may also be used to coordinate the robot swarm. In this paper, we describe an approach to the dual problems of swarm coordination and charging using a programmable light field, and demonstrate the efficacy of this approach in completing the 2D construction task.

Preprint submitted to 2019 International Conference on Robotics and Automation (ICRA). Received September 15, 2018.

II. Related Work

Early multi-robot object manipulation work takes inspiration from collective transport in ants and termites. Kube and Bonabeau [5] and Stilwell and Bay [6] both use small teams of robots to transport one object across an arena. Kube and Bonabeau use phototaxis and a light source to coordinate movement, but the source of light is affixed to the object to be pushed. This does not scale to constructing assemblies of many objects.

Rubenstein et al. [7] use a phototaxic swarm to move a single (potentially jointed and actively moving) object. By moving towards a light bulb at the edge of the arena, non-communicative robots ensure that they collectively move the object in the right direct. Our approach extends this idea by using a swarm to move many objects for the express purpose of construction, and by dynamically moving the light source so that the swarm moves in a non-uniform pattern.

Gauci et al. [8] use phototaxis (and anti-phototaxis) to organize robots into a desired shape. Robots begin in a large rectangle configuration and upon sensing an overhead light, each robot individually decides if they should stay in the light or not. The algorithm Gauci et al. present relies on inter-robot communication in a fixed radius around each robot. Our algorithm requires no inter-robot communication except in the form of physical interference.

Ferrante et al. [9] use phototaxis and anti-phototaxis to help divide labor among a resource-collecting swarm.

[10], [11], [1], and [12] all investigate swarm construction using passive building materials. [11] employs a swarm to build a linear wall across an arena, while [1] and [12] build more general structures out of special components. [1] uses magnetized joints and beams to construct 3D cubic structures. In [12], the robots construct 3D structures out of tiles instrumented with RFID tags. In [13], Werfel and Nagpal provide an algorithm for constructing 3D shapes out of passive blocks that perform computation but are immobile. Our approach is similar to all these in that we use an active team of agents and a passive building material. We extend the above work by using simpler robots to construct more general shapes.

[2], [8], [14] and [15] all solve the swarm construction problem by constructing polygons out of the robots themselves. Although self-assembly is a mature topic, and could be tackled with our approach, it is not on the path to scalable micro-assembly due to a fundamental limitation: robots are unlikely to be the target building material of produced objects. Instead we consider a robot system that manipulates external passive atomic building blocks. As long as the robots have sufficient pushing power to move the material, our algorithm is as applicable to gathering boxes in a warehouse as to assembling organ cells in a petri dish.

To the best of our knowledge no previously described swarm controller has used a programmable light field to both charge and coordinate the robot swarm. We believe the merging of these paradigms to be the primary contribution of this paper.

III. Approach

By dilating and contracting a light field that outlines a desired polygon, we coordinate a population of phototaxic robots to push boxes and construct a shape. The light field both powers the robots and directs them to specific areas of the arena. We complete the 2D construction task completely open-loop; there is no inter-robot communication, no light-to-robot communication, no control feedback, and no robot control except via the pre-determined patterns of illumination. Our approach is scalable with respect to the number of agents, as the complexity of the light controller is a function of the number of vertices of the input polygon rather than the size of the swarm.

A. The Robot Controller

Our robots do not perform any computation beyond the simple phototaxis behaviour, which can be achieved with an analog implementation. The robots have four photocells on the tops of their bodies, located at each of the front, back, left and right sides. The robot has a small energy store such as a capacitor which is charged by light on the cells. The control scheme is simple, a trivial extension of the second Braitenburg Vehicle [16]: the robot will turn and drive forward or backward with speed proportional to the difference between the photocell voltages (or sensor readings), front from back and left from right. This results in robots moving up local light intensity gradients and stopping at local maxima. Moving discharges the energy store, so that a robot that is away from a light source for too long is immobile until the light field illuminates it again.

It is conceivable that phototaxic prokaryotes or bacteria [17][18] could be employed instead of robots, as some species exhibit phototaxis and the ability to push a passive material.

B. The Light Field Controller

Algorithm 1 Light Controller to perform collective construction task

```
1:  procedure LIGHT FIELD UPDATE
2:      polygon ← Vertices specifying desired shape
3:      boxArea ← numBoxes * singleBoxArea
4:      minArea ← boxArea + robotSize
5:      for light in lightField do
6:          if polygon.distanceTo(light) < 1m then
7:              lightField(light) ← ON
8:      while running do
9:          while lightField.area() > minArea do
10:             lightField.shrink()
11:         if lightField.area() == minArea then
12:             wait for t worldSteps
13:         while lightField.extent() < arena.size() do
14:             lightField.grow()
```

Here we describe a dynamic light field programming strategy that is parameterized by the main properties of our

Preprint submitted to 2019 International Conference on Robotics and Automation (ICRA). Received September 15, 2018.

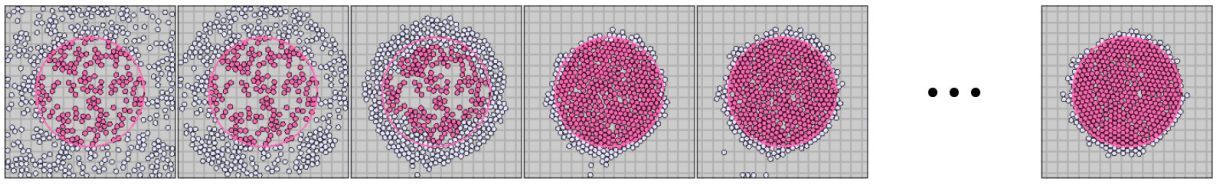


Fig. 2. Evolution of box positions over time. Robots are not shown. The leftmost image shows the initial uniform-at-random placement of boxes, followed by the movement of the boxes across the first complete compression cycle of the light field. The rightmost image shows the constructed shape after 100,000 world steps (7 full compression cycles). Boxes are colored pink when contained by the goal shape.

setting. The light controller is adaptive to the size and number of the building blocks, and the size of the robots, which are supplied before run-time as constant parameters. Using these values, the controller can determine the minimum size the light field can contract to while still retaining space for the boxes. Ideally, all boxes become optimally tiled, with this tessellation outlined by the light field's minimum level of contraction. Since our controller is adaptive, the same algorithm can be used with any number of robots and boxes of any shape or size. Throughout this paper, we refer to the polygon that best fits a tight packing of all boxes in the arena as the 'goal' or 'goal shape'.

The light controller calculates a minimum and maximum size that it should contract and dilate to (lines 2-7 of Algorithm 1), and will continually contract while the light field is larger than the minimum (lines 9-10), hold at the minimum for t world steps[1] (lines 12) and grow while the light field is smaller than its maximum size (lines 13-14). The algorithm halts when a maximum running time is reached.

C. The Simulator

We implemented our simulation using the Box2D physics engine[2]. Robots and boxes have density, friction and restitution properties. Table 1 shows the values used in our experiments. In Box2D, friction is measured from 0 to 1, where 0 represents a frictionless world. Restitution similarly is measured from 0 to 1, where 0 represents all of the energy from an impact being transfered to the impacted object, and 1 represents all the energy from a bounce being conserved.

TABLE I
PHYSICAL PROPERTIES OF SIMULATION IN BOX2D

Box Density	0.5 kg/m^2
Box Friction	1.0
Box Restitution	0.1
Robot Density	10 kg/m^2
Robot Friction	1.0
Robot Restitution	0

IV. EXPERIMENT

A. Setting

Robots are non-holonomic wheeled drive, homogenous, circular, and have a radius of 0.25m: roughly the size of an iRobot create. Other sizes are chosen in proportion to this base robot size. We call the atomic construction blocks 'boxes'. Boxes are homogeneous and hexagonal for convenient packing, but there is no special reason that this shape is necessary. The boxes have an apothem of 0.25m.

The arena is 64m x 64m across all tests. The lights in the overhead grid are uniformly spaced by 1m, modeled as point sources hanging 1m above the arena. We calculate the luminosity Lum_{xy} at a particular point as follows, where $Dist_{xy}$ and $Angle_{xy}$ are the distance and angle respectively from that point (x, y) on the arena floor to a light source L_i.

$$Brightness_{xy} = \frac{Intensity(L_i)}{Dist_{xy}} \quad (1)$$

$$Lum_{xy} = \sum_{\text{lights in range}} Brightness_{xy} \cdot sin(Angle_{xy}) \quad (2)$$

Equation (1) models light propagation where intensity decreases with respect to distance from the source. Equation (2) allows a single location to receive luminosity from potentially many light sources.

Robots are placed uniformly at random in a 32m x 32m square centered in the arena, outside the construction area (described below). We place the robots in such a restricted space compared to the arena to account for the dilation of the light field. In a smaller arena, concave light fields will not be able to dilate sufficiently to re-space the robots while still keeping all their vertices inside the arena. This has the undesirable effect of bunching robots on the vertices that stay inside the arena, since phototaxic robots will travel down the light that stays inside the arena.

Boxes are placed at random in 16m x 16m construction area that is delimited by a robot-permeable but box-impermeable wall. This wall ensures that robots do not drag boxes towards the corners of the arena upon dilation. This could be instantiated in the real world using a barrier for which robots are short enough to pass under, but boxes are not. The inclusion of this wall represents a limitation to our approach, and we hope to develop a controller that makes this unnecessary in the future. We refer to boxes outside of the goal as 'non-goal boxes' throughout for brevity.

In our tests we use 200 robots and 500 boxes. Small changes in populations yield small changes in results. Optimizing these parameters is beyond the scope of this paper.

B. Experiments

To demonstrate the approach, we tested several iterations of light field design, each with different goal shapes, repeated

[1] $t = 25$ in our experiments
[2] `https://box2d.org/`

Preprint submitted to 2019 International Conference on Robotics and Automation (ICRA). Received September 15, 2018.

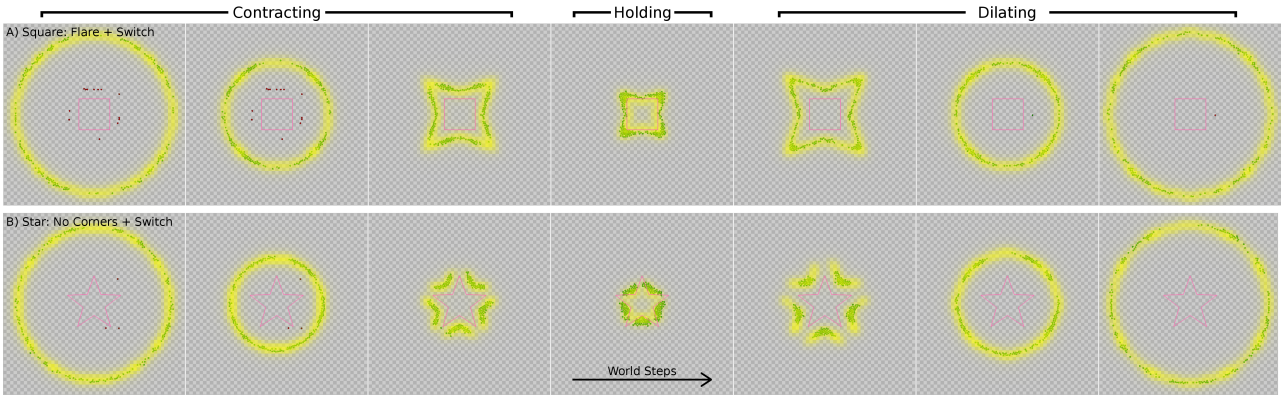


Fig. 3. One full period of a contraction, hold, and dilation cycle of the light-controller for square-shaped goals (A) and star-shaped goals (B). In both cases, we show the most sophisticated strategies we used (in the square case, flaring the corners and switching to a circle to re-spread; for the star, turning off the lights near the points, and switching to a circle to re-spread robots). We omit boxes to highlight the effect of the light on robot movement. See later figures for the boxes.

over multiple initial object placements. We use the following metrics to measure the success of an experiment:

1) Percentage of boxes inside the goal (high is good).
2) Average distance of all non-goal boxes to the closest point inside the goal shape (low is good).

The second metric allows us to rank results that achieve the same percentage of goal-boxes. Intuitively, we prefer that boxes that do not make it to the goal to be as close as possible to it.

The most simple test case uses a circular light field. This is the default if no polygon is specified (all lights that are a changing radius away from the center of convergence are turned on; no vertices defining the polygon are needed).

We also run tests on a square-shaped goal, and a star-shaped goal. We noticed that with the bare strategy, robots tend to clump close to the corners of the square, resulting in a rounded square construction. We remedied this by adding a new vertex at the midpoint of each line of the square light field, and flaring out the corners to account for the clumping (as seen in Fig. 3a). This resulted in sharper corners on square goal shapes, but spread the non-goal boxes farther from the goal. In flaring the corners, we transform the square into a new concave polygon.

C. Fixing Concave Shapes

During experimentation, we noticed that concave shapes tended to cluster robots around the vertices farthest away from the center of convergence. Depending on the shape, this either rounded the corners, or made certain non-goal boxes less likely to be reached as the robot-spacing became less uniform. We addressed this problem by switching to a circle-shaped light field once the light field had dilated to 25% of its total dilation. The light field returned to the desired shape once it contracted back down to this same point. This forced robots to space themselves more uniformly around the light field since the robots interfere with each other and compete for space in the light field. The 'Switch' cases in our figures use this strategy (see Fig. 3).

We found one additional alteration to the control scheme that improved performance for the concave star shape. The sharp corners still tend to bunch robots even after switching, resulting in rounded stars. To remedy this, we turned off 50% of the lights that were closest to the convex vertices (shown in in Fig. 3b). We refer to this case as the 'No Corners' case. This strategy is especially attractive because it limits the amount of light energy needed to power the system. Future work could explore other patterns that change over time, and increase performance for certain classes of shapes.

V. Results

We repeated our experiments over starting positions of boxes and robots placed uniformly at random. The values for our figures are calculated as follows: For a given experiment, we measure the distances of all boxes every 1000 time steps. We find the percentage of boxes inside the goal shape, and the average distance of all non-goal boxes. After running all of our trials, we collect these values in a second pair of lists corresponding to each metric.

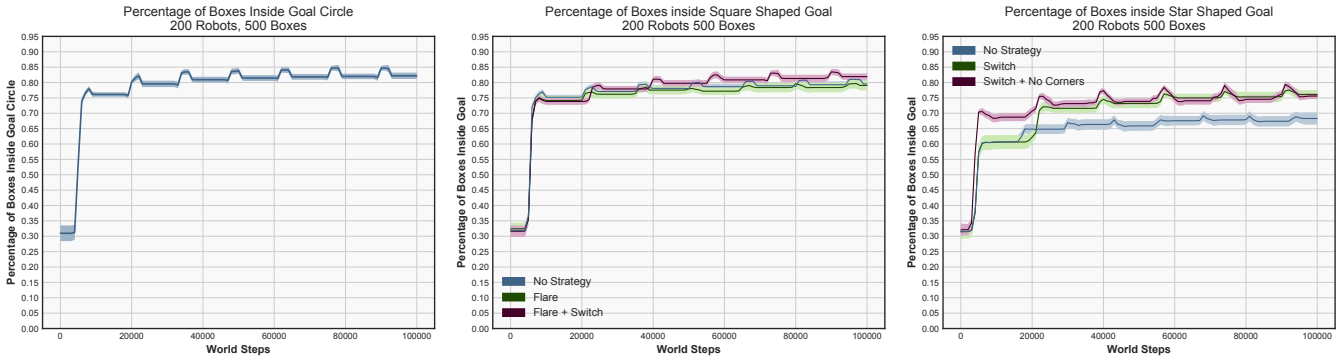
The mean we report then is the mean of the sampling distribution of the mean. Similarly the standard deviation we report is a measure of spread of the mean values across all trials. The shaded area around the data in each our figures represents one standard deviation above and below the data point.

A. Constructing Circles, Squares & Stars

Fig. 4a and Fig. 5a show how quickly the boxes converge in the circle setting. After a single compression cycle, around 75% of the boxes are in the right location, and the average distance away from the goal of non-goal boxes continues to decrease steadily after this initial jump. Upon halting, we achieve about 80% success, with the remaining boxes being on average half a meter (one box-length) away from the goal shape.

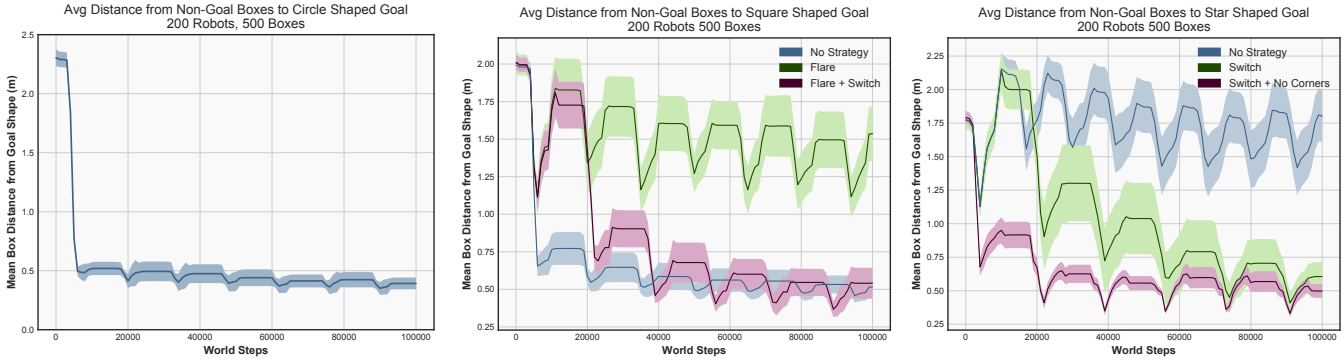
Note that the bumps seen in Fig. 4 delimit the complete compression and dilation cycles. Since robots tend to drag some boxes with them when pulling away from the shape, the

Preprint submitted to 2019 International Conference on Robotics and Automation (ICRA). Received September 15, 2018.



(a) Circle Shape (b) Square Shape, 3 Strategies (c) Star Shape, 3 Strategies

Fig. 4. Percentage of boxes reaching the goal when constructing circles, squares, and stars. The best possible controller would achieve a score of 1 (all boxes inside the goal), and no other tiling could be better than this tiling since at best every box contributes to a complete tessellation of the goal-shape.



(a) Circle Shape (b) Square Shape, 3 Strategies (c) Star Shape, 3 Strategies

Fig. 5. Average distance from non-goal boxes to the goal when constructing circles, squares, and stars. The optimal controller would achieve an average non-goal distance of 0m, indicating that all boxes were inside the goal shape. This is only possible in an optimal tiling since boxes are non-overlapping.

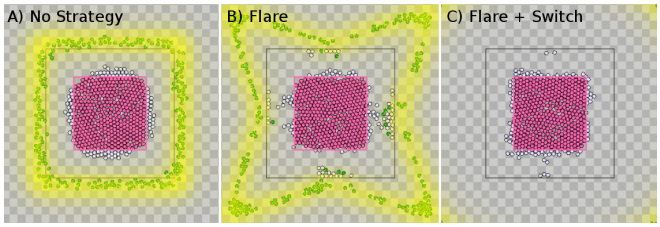


Fig. 6. A comparison after 100,000 world steps of the three strategies used to construct a square shape.

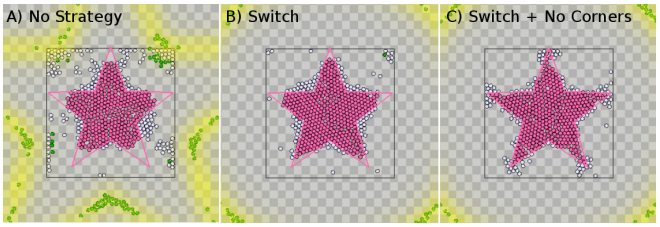


Fig. 7. A comparison after 100,000 world steps of the three strategies used to construct a star shape.

average distance of boxes to the goal will increase slightly when the light field begins to dilate away from its minimum size. This effect can be seen across our experiments.

We ran the same experiment with a square-shaped goal and measured the convergence of boxes to the goal (Fig. 4b and Fig. 5b). The controller achieves success similar to that exhibited in the circle case. An interesting result is the difference that switching makes. If robots are not spread evenly around the light field (as occurs for concave polygons), the non-goal boxes tend to be dragged farther away from the goal than if we had switched. Fig. 6 shows the improvement described here.

Fig. 4c and Fig. 5c show our results using a star-shaped goal. The performance is slightly worse than both the circle and square (about 5% less boxes reaching the goal at 100,000 steps), but still performs well given the relative complexity of the star shape. Again, we see that switching has a significant impact on the closeness of non-goal boxes to the goal shape. We also achieve success by turning off half the lights closest to the points of the star (the 'No Corners' case). Fig. 7 shows the shapes constructed by each of the three strategies.

B. *Statistics*

In the following tables we report our statistics at the one-hundred-thousandth world step. We ran 100 trials for the

circle case, and 40 trials for every other case.

TABLE II
STATISTICAL DATA: PERCENTAGE OF BOXES INSIDE GOAL

Case	Mean	Median	StdDev	StdErr
Circle	82.23%	82.2%	0.9387	0.0939
Square: No Strategy	79.47%	79.40%	1.176	0.1860
Square: Flare	79.04%	79.2%	1.594	0.2520
Square: Flare + Switch	81.92%	82.0%	1.023	0.1617
Star: No Strategy	68.295%	68.6%	1.979	0.3129
Star: Switch	76.09%	76.4%	1.455	0.2301
Star: Switch + No Corners	75.59%	75.8%	0.9874	0.1561

TABLE III
STATISTICAL DATA: DISTANCE OF NON-GOAL BOXES FROM GOAL

Case	Mean	Median	StdDev	StdErr
Circle	0.3922m	0.3844m	0.0499	0.0050
Square: No Strategy	0.5150m	0.5177m	0.0573	0.0091
Square: Flare	1.536m	1.505m	0.1790	0.0283
Square: Flare + Switch	0.5406m	0.5349m	0.1028	0.0163
Star: No Strategy	1.802m	1.751m	0.1917	0.0303
Star: Switch	0.6057m	0.5990m	0.1072	0.0170
Star: Switch + No Corners	0.4979m	0.4867m	0.0513	0.0081

TABLE IV
2-SAMPLE KOLMOGOROV-SMIRNOV SIGNIFICANCE - PERCENTAGE OF BOXES INSIDE GOAL

Case	D-Stat	P-Value
Square: No Strategy & Flare	0.41584	$p << 0.01$
Square: Flare & Flare + Switch	0.60396	$p << 0.01$
Square: No Strategy & Flare + Switch	0.47525	$p << 0.01$
Star: No Strategy & Switch	0.78218	$p << 0.01$
Star: Switch & Switch + No Corner	0.19801	0.03272
Star: No Strategy & Switch + No Corner	0.91089	$p << 0.01$

TABLE V
2-SAMPLE KOLMOGOROV-SMIRNOV SIGNIFICANCE - DISTANCE OF NON-GOAL BOXES FROM GOAL

Case	D-Stat	P-Value
Square: No Strategy & Flare	0.94059	$p << 0.01$
Square: Flare & Flare + Switch	0.80198	$p << 0.01$
Square: No Strategy & Flare + Switch	0.29703	0.00019
Star: No Strategy & Switch	0.79207	$p << 0.01$
Star: Switch & Switch + No Corner	0.57426	$p << 0.01$
Star: No Strategy & Switch + No Corner	0.96040	$p << 0.01$

VI. DISCUSSION

A. Limitations

Our results rely on the assumption that robots will primarily be pushing objects that are of comparable size to themselves. We have not measured how robot size interacts with box size, nor have we tested different widths of light fields, using two or more layers of lights to outline the shape. These represent experiments that would be appropriate for future papers. Our large parameter space means there is much optimization work that could be done to describe the ideal proportions between box and robot size, as well as box and robot population size.

Here we used a popular third-party physics model for simulation. We do not yet have arguments about the necessary and sufficient density, friction and restitution of the robots and boxes.

Finally, while our controller is scalable, the physics engine we use is not. Experiments with thousands of objects would take an inconveniently long time to run. While this is not a limitation of our controller, it does create a challenge for running larger experiments.

B. Future Work

Our key contribution is the idea of using a programmable light field to simultaneously power and control phototaxic robots. We chose to do open-loop control for simplicity. Even while maintaining open-loop light control, there are many variations that can be considered. Further, we can close the light control loop around either the current state of the boxes or the robots. This requires sensing the state, but could offer far better control.

Orthogonally, we can consider heterogeneous swarms that react differently depending on the frequency of light, or that have different behaviors in the same light, perhaps allowing one to organize a swarm into groups. The controller could be extended to coordinate the movement of objects upon construction, or to construct second-order objects (using the earlier objects as component parts of later objects). Each of these ideas could be implemented using our existing simulation as a base.

To facilitate reproduction and extension, we have made our code public, and include the git hash reflecting the state of the code at the time of this paper's writing[3].

An interesting extension would be to examine pushing aggregate materials, so the robots are more like bulldozers than box-pushers. The most exciting challenge would be to attempt an implementation using phototaxic organisms.

VII. CONCLUSION

The contributions of this paper include: (a) the novel strategy of using a light field to both power and control a robot swarm, (b) a simple algorithm that generates light field patterns that can be played open-loop to drive phototaxic robots to construct user-defined polygons, and (c) a simple physics-based box-pushing simulator and implementations of the algorithms for further experiments.

We have shown empirically that our controller can successfully push boxes to construct circles, squares, and stars, and analyzed some of the limitations of our approach.

ACKNOWLEDGEMENT

This work is supported by an NSERC USRA and the NSERC Canadian Field Robotics Network.

[3]https://github.com/AdamBignell/push
SHA-1 Hash: 6c790b0c995facfc2723e8fb866d6bdb2130afca

REFERENCES

[1] Q. Lindsey, D. Mellinger, and V. Kumar, "Construction of cubic structures with quadrotor teams," in *Proceedings of Robotics: Science and Systems*, vol. 7, 01 2011, pp. 177–184.

[2] M. Rubenstein, A. Cornejo, and R. Nagpal, "Programmable self-assembly in a thousand-robot swarm," *Science*, vol. 345, no. 6198, pp. 795–799, 2014. [Online]. Available: http://science.sciencemag.org/content/345/6198/795

[3] R. Vaughan, *36 iRobot Create Robots clustering boxes, 10 x speed*, 2007. [Online]. Available: https://www.youtube.com/watch?v=b_kZmatqAaQ

[4] M. Rubenstein, C. Ahler, N. Hoff, A. Cabrera, and R. Nagpal, "Kilobot: A low cost robot with scalable operations designed for collective behaviors," *Robotics and Autonomous Systems*, vol. 62, no. 7, pp. 966 – 975, 2014, reconfigurable Modular Robotics. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0921889013001474

[5] C. Kube and E. Bonabeau, "Cooperative transport by ants and robots," *Robotics and Autonomous Systems*, vol. 30, no. 1, pp. 85 – 101, 2000. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0921889099000664

[6] D. J. Stilwell and J. S. Bay, "Toward the development of a material transport system using swarms of ant-like robots," in *[1993] Proceedings IEEE International Conference on Robotics and Automation*, May 1993, pp. 766–771 vol.1.

[7] M. Rubenstein, A. Cabrera, J. Werfel, G. Habibi, J. McLurkin, and R. Nagpal, "Collective transport of complex objects by simple robots: Theory and experiments," in *Proceedings of the 2013 International Conference on Autonomous Agents and Multi-agent Systems*, ser. AAMAS '13. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2013, pp. 47–54. [Online]. Available: http://dl.acm.org/citation.cfm?id=2484920.2484932

[8] M. Gauci, R. Nagpal, and M. Rubenstein, "Programmable self-disassembly for shape formation in large-scale robot collectives," in *Distributed Autonomous Robotic Systems: The 13th International Symposium*, R. Groß, A. Kolling, S. Berman, E. Frazzoli, A. Martinoli, F. Matsuno, and M. Gauci, Eds. Cham: Springer International Publishing, 2018, pp. 573–586. [Online]. Available: https://doi.org/10.1007/978-3-319-73008-0_40

[9] E. Ferrante, A. E. Turgut, E. Duez-Guzmn, M. Dorigo, and T. Wenseleers, "Evolution of self-organized task specialization in robot swarms," *PLOS Computational Biology*, vol. 11, no. 8, pp. 1–21, 08 2015. [Online]. Available: https://doi.org/10.1371/journal.pcbi.1004273

[10] J. Werfel, "Collective construction with robot swarms," in *Morphogenetic Engineering: Toward Programmable Complex Systems*, R. Doursat, H. Sayama, and O. Michel, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 115–140. [Online]. Available: https://doi.org/10.1007/978-3-642-33902-8_5

[11] Z. Zhang, Y. Li, and C. Li, "Modelling of collective construction in a minimalist robotic swarm," in *Advances in Computer Science, Environment, Ecoinformatics, and Education*, S. Lin and X. Huang, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 385–391.

[12] J. Werfel, K. Petersen, and R. Nagpal, "Designing collective behavior in a termite-inspired robot construction team," *Science*, vol. 343, no. 6172, pp. 754–758, 2014. [Online]. Available: http://science.sciencemag.org/content/343/6172/754

[13] J. Werfel and R. Nagpal, "Three-dimensional construction with mobile robots and modular blocks," *Int. J. Rob. Res.*, vol. 27, no. 3-4, pp. 463–479, Mar. 2008. [Online]. Available: http://dx.doi.org/10.1177/0278364907084984

[14] E. Klavins, S. Burden, and N. Napp, "Optimal rules for programmed stochastic self-assembly," in *Robotics: Science and Systems*, 2006.

[15] R. Gross, M. Bonani, F. Mondada, and M. Dorigo, "Autonomous self-assembly in swarm-bots," *IEEE Transactions on Robotics*, vol. 22, no. 6, pp. 1115–1130, Dec 2006.

[16] V. Braitenberg, *Vehicles: Experiments in Synthetic Psychology*, ser. Bradford Books. MIT Press, 1986. [Online]. Available: https://books.google.com/books?id=7KkUAT_q_sQC

[17] W. D. Hoff, M. A. van der Horst, C. B. Nudel, and K. J. Hellingwerf, "Prokaryotic phototaxis," in *Chemotaxis: Methods and Protocols*, T. Jin and D. Hereld, Eds. Totowa, NJ: Humana Press, 2009, pp. 25–49. [Online]. Available: https://doi.org/10.1007/978-1-60761-198-1_2

[18] G. Jékely, "Evolution of phototaxis," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 364, no. 1531, pp. 2795–2808, 2009. [Online]. Available: http://rstb.royalsocietypublishing.org/content/364/1531/2795