

only map of any DIB carrier at this scale and the only one taking the distance information as a major parameter. Together with the measured scale height, this is the first 3D study of the spatial distribution of the DIB-bearing ISM clouds. The projected distribution of the extinction due to the interstellar dust is markedly similar to that of the DIB carrier [see (26) for the correlation analysis], confirming the strong correspondence between the two (32). The map of the extinction is itself an advance, as it maps the regions out of the Galactic plane and probes dust to greater distances than present maps (33) of these regions and is consistent with maps in the literature. Our success in producing the maps of the DIB carrier implies good prospects for future spectroscopic surveys (14–16) that will produce similar (15) or better quality (14, 16) spectra and will also rely on DIBs to provide information about the ISM. Our work opens new possibilities in the study of DIBs and also offers a unique way of comparing DIBs with other interstellar species by studying their out-of-plane distribution. This can be translated into the study of physical and chemical properties of DIB carriers in the near future.

The measured 3D distribution, especially the unexpectedly high scale-height of the DIB 8620 carrier, calls for a theoretical explanation. There are two options—either the DIB carriers migrate to their observed distances from the Galactic plane, or they are created at these large distances, from components of the ISM having a similar distribution. The latter is simpler to discuss, as it does not require knowledge of the chemistry of the DIB carrier or processes in which the carriers are involved. Khoperskov and Shchekinov (34) showed that mechanisms responsible for dust migration to high altitudes above the Galactic plane segregate small dust particles from large ones, so the small ones form a thicker disk. This is also consistent with the observations of the extinction and reddening at high Galactic latitudes (35).

REFERENCES AND NOTES

1. G. H. Herbig, *Annu. Rev. Astron. Astrophys.* **33**, 19–73 (1995).
2. P. J. Sarre, *J. Mol. Spectrosc.* **238**, 1–10 (2006).
3. L. M. Hobbs et al., *Astrophys. J.* **705**, 32–45 (2009).
4. G. Galazutdinov, B.-C. Lee, I.-O. Song, M. Kazmierczak, J. Kretowski, *Mon. Not. R. Astron. Soc.* **412**, 1259 (2011).
5. J. Kretowski et al., *Astrophys. J.* **714**, L64 (2010).
6. F. Salama, G. A. Galazutdinov, J. Kretowski, L. J. Allamandola, F. A. Musaev, *Astrophys. J.* **526**, 265–273 (1999).
7. S. Iglesias-Groth et al., *Mon. Not. R. Astron. Soc.* **407**, 2157–2165 (2010).
8. J. P. Maier et al., *Astrophys. J.* **726**, 41 (2011).
9. J. A. Thorburn et al., *Astrophys. J.* **584**, 339–356 (2003).
10. D. Milisavljevic et al., *Astrophys. J.* **782**, L5 (2014).
11. A. G. G. M. Tielens, *The Physics and Chemistry of the Interstellar Medium* (Cambridge Univ. Press, Cambridge, 2005).
12. G. Kordopatis et al., *Astron. J.* **146**, 134 (2013).
13. D. J. Eisenstein et al., *Astron. J.* **142**, 72 (2011).
14. G. Gilmore et al., *Messenger* **147**, 25–31 (2012).
15. J. H. J. de Bruijne, *Astrophys. Space Sci.* **341**, 31–41 (2012).
16. K. C. Freeman, *Galactic Archaeology: Near-Field Cosmology and the Formation of the Milky Way*. W. Aoki, M. Ishigaki, T. Suda, T. Tsujimoto, N. Arimoto, Eds. (2012), vol. 458 of *Astronomical Society of the Pacific Conference Series*, p. 393.
17. L.-C. Deng et al., *Res. Astron. Astrophys.* **12**, 735–754 (2012).

18. T. P. Snow Jr., D. G. York, D. E. Welty, *Astron. J.* **82**, 113 (1977).
19. J. T. van Loon et al., *Astron. Astrophys.* **550**, A108 (2013).
20. S. D. Friedman et al., *Astrophys. J.* **727**, 33 (2011).
21. J. Kos, T. Zwitter, *Astrophys. J.* **774**, 72 (2013).
22. J. Kos et al., *Astrophys. J.* **778**, 86 (2013).
23. H.-C. Chen et al., *Astron. Astrophys.* **550**, A62 (2013).
24. J. Binney et al., *Mon. Not. R. Astron. Soc.* **437**, 351–370 (2014).
25. D. J. Schlegel, D. P. Finkbeiner, M. Davis, *Astrophys. J.* **500**, 525–553 (1998).
26. Supplementary materials are available on Science Online.
27. D. B. Sanders, P. M. Solomon, N. Z. Scoville, *Astrophys. J.* **276**, 182 (1984).
28. J. G. A. Wouterloot, J. Brand, W. B. Burton, K. K. Kwee, *Astron. Astrophys.* **230**, 21 (1990).
29. K. R. Sembach, A. C. Danks, *Astron. Astrophys.* **289**, 539 (1994).
30. J. M. Dickey, F. J. Lockman, *Annu. Rev. Astron. Astrophys.* **28**, 215–259 (1990).
31. S. Sharma, J. Bland-Hawthorn, K. V. Johnston, J. Binney, *Astrophys. J.* **730**, 3 (2011).
32. U. Munari et al., *Astron. Astrophys.* **488**, 969–973 (2008).
33. G. A. Gontcharov, *Astron. Lett.* **36**, 584–595 (2010).
34. S. A. Khoperskov, Y. A. Shchekinov, Transport of charged dust grains into the galactic halo (2014); <http://arxiv.org/abs/1403.7075>.
35. J. E. G. Peek, Ultraviolet extinction at high galactic latitudes II: The ultraviolet extinction function (2013); <http://arxiv.org/abs/1311.2941>.

ACKNOWLEDGMENTS

We are grateful to the anonymous referees for valuable advice on the discussion of possible DIB origins. J.K. and T.Z. thank A. F. Kord for a stimulating discussion of the topic. Funding for RAVE has been provided by the Anglo-Australian Observatory; the Leibniz-Institut für Astrophysik Potsdam; the Australian National University; the Australian Research Council; the French National Research Agency; the German Research Foundation (SFB 881); the Istituto Nazionale di Astrofisica di Padova; The Johns Hopkins University; the U.S. National Science Foundation (AST-0908326); the W. M. Keck foundation; the Macquarie University; the Netherlands Research School for Astronomy; the Natural Sciences and Engineering Research Council of Canada; the Slovenian Research Agency; Center of Excellence Space.si; the Swiss National Science Foundation; the Science and Technology Facilities Council of the UK; Opticon; Strasbourg Observatory; European Research Council; and the Universities of Groningen, Heidelberg, and Sydney. The RAVE Web site is at www.rave-survey.org.

SUPPLEMENTARY MATERIALS

www.sciencemag.org/content/345/6198/791/suppl/DC1
Materials and Methods
Supplementary Text
Figs. S1 to S3
References
10 March 2014; accepted 19 June 2014
10.1126/science.1253171

ROBOTICS

Programmable self-assembly in a thousand-robot swarm

Michael Rubenstein,* Alejandro Cornejo, Radhika Nagpal

Self-assembly enables nature to build complex forms, from multicellular organisms to complex animal structures such as flocks of birds, through the interaction of vast numbers of limited and unreliable individuals. Creating this ability in engineered systems poses challenges in the design of both algorithms and physical systems that can operate at such scales. We report a system that demonstrates programmable self-assembly of complex two-dimensional shapes with a thousand-robot swarm. This was enabled by creating autonomous robots designed to operate in large groups and to cooperate through local interactions and by developing a collective algorithm for shape formation that is highly robust to the variability and error characteristic of large-scale decentralized systems. This work advances the aim of creating artificial swarms with the capabilities of natural ones.

In nature, groups of thousands, millions, or trillions of individual elements can self-assemble into a wide variety of forms, purely through local interactions. Examples can be found across a wide range of physical scales and systems: at the molecular scale with self-assembly of crystals or rotary motors of bacterial flagella (1, 2), at the cellular scale with the development of multicellular organisms (3, 4), and at the colony level with ants creating structures such as rafts, chains, and nests (bivouacs) using only their interconnected bodies as building material (5, 6). Through collective shape formation, a group can dramatically change how it interacts with its environment. For example, the evolution of multicellular

body plans enabled organisms to rapidly fill many ecological niches (7), and self-assembly of bridges and bivouacs allows army ant colonies to traverse difficult terrain while providing security and environmental regulation for the queen and brood (5). These examples of collective intelligence are fascinating to scientists across disciplines, as much of the global complexity arises from interactions among individuals that are myopic, sensing and interacting at scales many orders of magnitude smaller than the phenomenon itself.

In the field of robotics, researchers use inspiration from collective intelligence in nature to create artificial systems with capabilities observed in natural swarms. Researchers have designed tiny robots, inspired by bees and ants, that are envisioned to work together in large groups, even assembling together to cross difficult terrains (8–10). Similarly, using inspiration from multicellular development, several groups have

School of Engineering and Applied Sciences and Wyss Institute for Biologically Inspired Engineering, Harvard University, Cambridge, MA 02138, USA.

*Corresponding author. E-mail: mrubenstein@seas.harvard.edu

developed self-assembling “cellular” robots that can reconfigure into many morphologies to complete different grasping or locomotion tasks (*11*), or can act as programmable matter that can rapidly assemble into user-specified tools such as a wrench or key (*12, 13*). However, there still exists a substantial gap between the conceptual designs and the realized systems. Increasing the scale of autonomous robotic systems remains a significant challenge both for algorithm and hardware design. Current systems are on the order of 10 to 50 robots, with only a few exceeding 100 robots (*14, 15*), in part because of cost and design choices that make manufacturing and operations difficult. This lack of large-scale physical systems has in turn hindered experimental investigation of algorithms for engineering self-assembly and other forms of complex collective intelligence. Although simulators are a valuable tool for systematically exploring algorithm behavior, they frequently involve simplifications for computational tractability. Such simulated systems can fail to faithfully reproduce the intricate physical interactions and variability that exist in real systems, and their fidelity to the real world is difficult to verify or improve without feedback from physical experiments.

We report a system that addresses both the physical and algorithmic challenges of robotic swarms. We demonstrate a thousand-robot swarm capable of large-scale, flexible self-assembly of two-dimensional shapes entirely through programmable local interactions and local sensing, achieving highly complex collective behavior. The approach involves the design of a collective

algorithm that relies on the composition of basic collective behaviors and cooperative monitoring for errors to achieve versatile and robust group behavior, combined with an unconventional physical robot design that enabled the creation of more than 1000 autonomous robots.

In designing a large-scale robot swarm, the extent to which the robots can be fully autonomous (capable of computation, locomotion, sensing, and communication) must be balanced against the cost per robot. Mass production favors robots with fewer and cheaper components, resulting in lower cost but also reduced capabilities and reliability. Additionally, operations such as powering on/off, reprogramming, and charging must be addressed; at the scale of a thousand robots, there is a substantial and practical need for group-level rather than individual-level operation (*16*). In general, how individual capabilities and reliability relate to collective capabilities and reliability remains an active theoretical question. Here, we demonstrate that the limited capabilities of our system are sufficient to permit the execution of a complex algorithm for self-assembly at the swarm level for a collective of 1024 robots (Fig. 1), both theoretically and experimentally.

We designed a simple low-cost robot called Kilobot, first presented in (*17*) with a small prototype swarm of 30 robots executing a suite of simple collective behaviors. Kilobot exploits several unconventional design choices, such as using vibration motors for sliding movement rather than traditional wheels (*18*) and reflecting infrared

light off the table surface below for communication and distance sensing between robots. Vibration motors provide noisy locomotion without position feedback, preventing a single robot from traveling long distances with any precision. Robots can measure distance to a neighboring robot by sensing the infrared light intensity of received messages, although this measurement is noisy and provides no information about the relative angle between the two robots. In addition, the system as a whole is inherently decentralized and asynchronous; any collective behavior must be achieved purely through local interactions between neighboring robots. Some individual limitations can be overcome through cooperation; a robot can use distance measurements from a stationary neighbor as feedback to enable more precise motion, and errors in distance sensing can be reduced by comparing readings among neighbors.

We developed a collective algorithm that guarantees that a large group of robots, with limited capabilities and local communication, can cooperatively assemble into any 1-connected user-specified shape [with some restrictions (*19*)]. The self-assembly algorithm (Fig. 2) composes three primitive collective behaviors: (i) edge-following, where a robot can move along the edge of a group by measuring distances from robots on the edge; (ii) gradient formation, where a source robot can generate a gradient value message that increments as it propagates through the swarm, giving each robot a geodesic distance from the source; and (iii) localization, where robots can form a

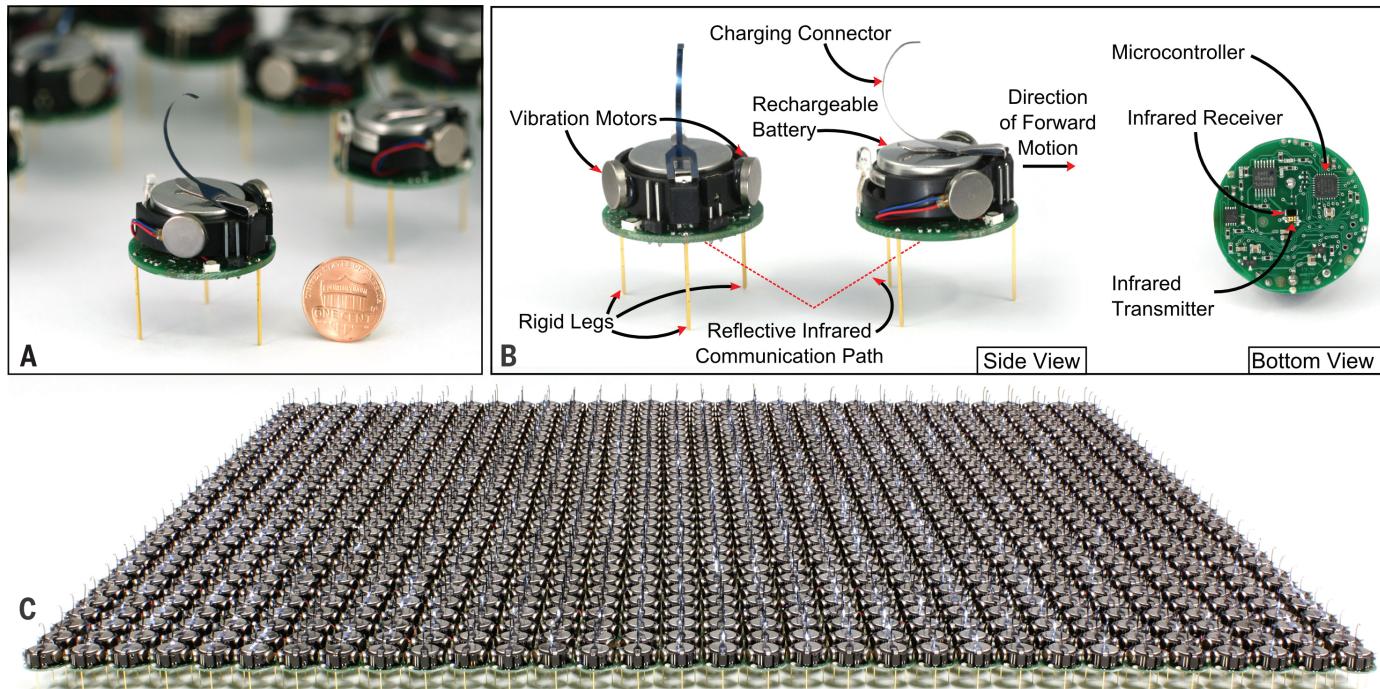


Fig. 1 Kilobot swarm robot. (A) A Kilobot robot, shown alongside a U.S. penny for scale. (B) Each Kilobot has an onboard microcontroller for executing programs autonomously, two vibration motors for moving straight or turning on a flat surface, and a downward-facing infrared transmitter and receiver. Robots communicate with others within a range of 10 cm (roughly three robot diameters) by reflecting infrared light off the table below. Communicating robots can evaluate relative distance by measuring the strength of the received infrared signal, but they cannot sense relative bearing (angle). (C) A 2^{10} Kilobot swarm. The Kilobot design allows for all operations on the entire swarm (charging, programming, etc.) to take a constant time to complete, independent of the number of robots in the swarm.

local coordinate system using communication with, and measured distances to, neighbors. The self-assembly algorithm links these three primitives with a finite-state automaton. We describe the algorithm at a high level here; see (19) for a detailed description of each aspect along with proofs of correctness and completion.

All robots are given an identical program that includes the fixed self-assembly algorithm and an image of the specific target shape and size. The robots are initially tightly packed into an arbitrarily shaped group without any knowledge about their location in the environment (Fig. 2). To start the self-assembly process, a user places four specially programmed seed robots next to the initial group, marking the position and orientation for where the shape should be formed in the environment. The seed robots become the source of a gradient formation that propagates through the initial group. Robots along the outer edge of the initial group are able to move without being physically blocked. They can determine that they are on the outer edge by comparing their gradient values to those of their neighbors. These robots can then use edge-following around the stationary initial group to reach the seed. To avoid congestion that would occur if all robots along the outer edge moved at once, random-

ness is used to allow only a subset to start motion at any one time. This method creates the effect of the initial group “dissolving” from the outer edge inward, with all robots eventually edge-following to the seed.

Robots have no direct information about their global position, only the distances to nearby neighbors. However, they can use this distance information to collectively construct a coordinate system. Initially, the stationary seed robots embody the origin of this coordinate system. As new robots arrive at the seed, they can use the localization primitive to determine their own position in this coordinate system; once this occurs, they can become reference points for the localization of additional robots. In this way, a coherent coordinate system can be built up through local interactions. The localization primitive works in the following way: Localized robots continually transmit messages that contain their (x, y) position in the coordinate system. A new robot listens and measures distances to all previously localized neighbors; if it can observe at least three stationary and noncollinear localized robots, it can compute its own (x, y) position using a distributed implementation of trilateration (Fig. 2) (19, 20). Moving robots continually compute both their location and gradient value as they move.

Once a robot is localized, it can determine whether it is located inside the desired shape by comparing its position to the shape image. If it is not within the shape, the robot edge-follows the partially formed assembly until it detects that it has entered the shape. Once inside the shape, the robot continues to edge-follow until one of two conditions is met. If the robot is about to exit the shape, or if it is next to (less than a predefined distance from) a stationary robot that has the same gradient value as itself, it becomes stationary for the remainder of the process. This process of robots moving into and joining the assembly continues until the shape is filled with robots. The overall effect is that the shape builds up in successive layers (Fig. 2 and Fig. 3J).

This seed-initiated self-assembly algorithm can form a large class of shapes, relying only on local interactions of the robots [see (19) for proof]. The algorithm resembles previous lattice-based algorithms for collective construction and self-assembly of modular robots (12, 13, 21–24) and uses conceptual primitive behaviors found in previous work, such as edge-following (23–25), gradient formation (23, 26) and localization (22, 23). However, unlike most previous methods, our algorithm allows components to be positioned freely

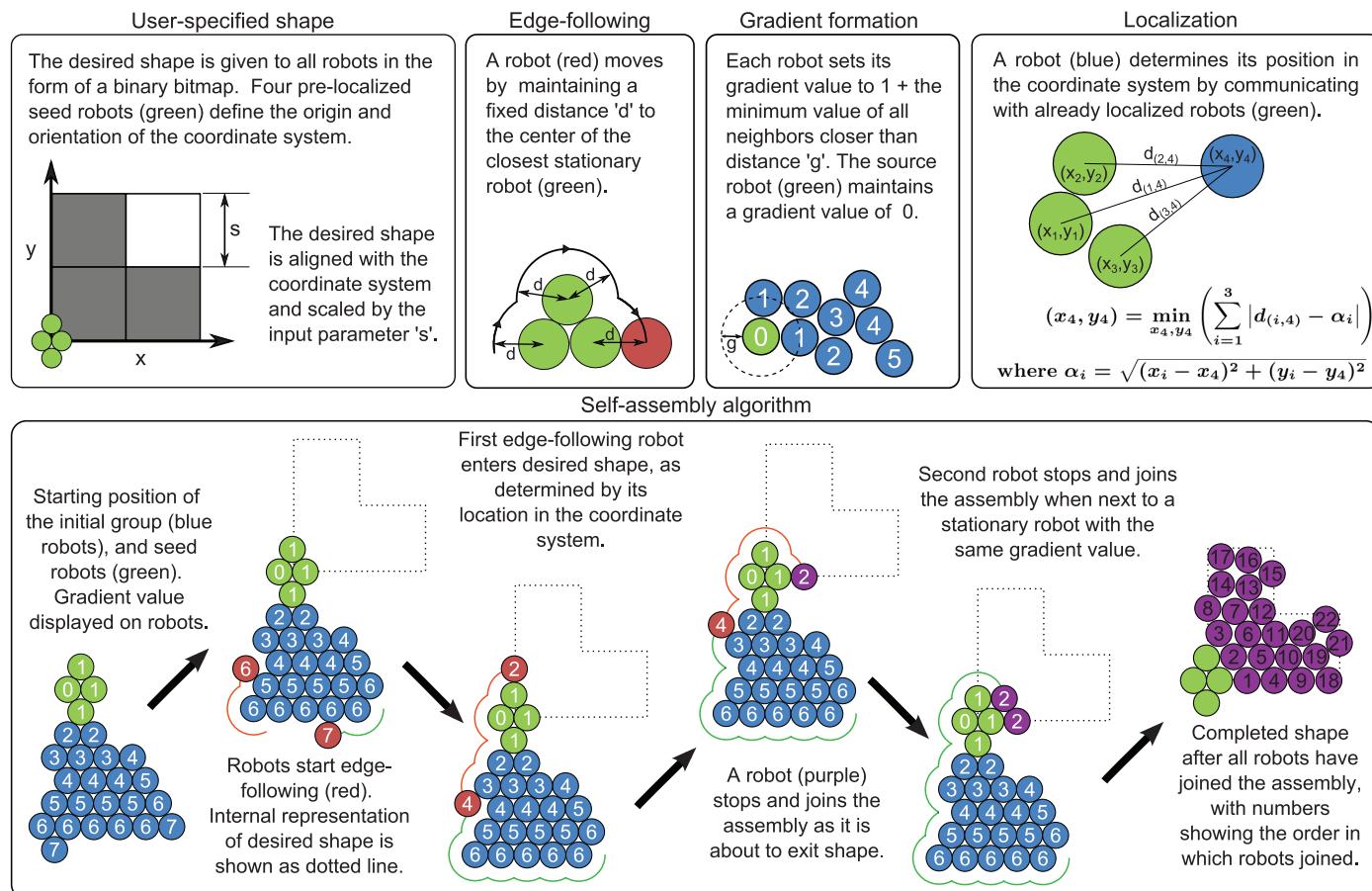


Fig. 2. Collective self-assembly algorithm. Top left: A user-specified shape is given to robots in the form of a picture. Top right: The algorithm relies on three primitive collective behaviors: edge-following, gradient formation, and localization. Bottom: The self-assembly process by which a group of robots forms the user-defined shape.

in continuous two-dimensional space and comes with a provable guarantee that the shape will form correctly. The use of continuous space implies that the implementations of the primitive behaviors are substantially different because of continuous and noisy sensing and motion, and proving correctness is more complex as the shape may be self-assembled differently each time because of differential packing of robots. However, avoiding grid restrictions allows for simpler robots and more robustness to imprecision in element positioning, which was instrumental in translating our algorithm onto a 1000-robot swarm.

Moving from abstract algorithms to physical robots requires solving several additional challenges. Mathematical models assume idealized behavior in order to achieve tractability; however, real robots exhibit much more complex behaviors. For example, robots often have imprecise locomotion, noisy distance sensing, and message loss. Moving from tens of robots to more than a thousand introduces additional nonideal behaviors that are not easily observed in smaller groups and are therefore rarely incorporated into simulators for modeling large robot swarms. These include systematic errors and high variability in robot locomotion and sensing, which can be manually detected and easily corrected in small groups but are too cumbersome and time-consuming to correct in larger groups, resulting in robots with statistically different variability. Increasing group size also increases the occurrence of statistically rare events, such as a robot becoming completely immobilized because of motor failure or a robot getting physically pushed by another. We observed that with a thousand physical robots running for tens of hours, high variability and rare errors are inevitable and capable of completely bringing such abstract algorithms to a halt.

To achieve robustness at the 1000-robot scale, we used several additional algorithmic strategies (19). For example, a continuous-space shape representation in our algorithm describes the desired shape as a boundary but does not dictate the exact location of robots within that shape. This enables the algorithm to tolerate a variety of packing patterns of robots within the shape, helping to prevent small position errors by the robots from propagating into large-scale self-assembly failures. One key technique for robustness was cooperative monitoring, in which a robot uses interactions with neighbors to detect and recover from faults that it cannot observe itself. For example, an immobilized robot can cause havoc in an edge-following algorithm by causing a permanent blockage. Similarly, localized robots can be physically pushed and therefore no longer have correct knowledge of their position in the swarm, and this error can then propagate through the rest of the assembling swarm. The robots lack the internal sensors to detect such rare events on their own; however, by using neighbor interaction, these faults can be detected. For example, both errors cause sensed distance between neighbors to change in predictable ways. This allows for

autonomous recovery: An immobilized robot can reset its motors and try again, or it can signal to other robots that they should move around it; similarly, a pushed robot can selectively reevaluate its position and correct for the unintended movement. In general, many faults can be detected through information exchange with neighbors; this cooperative monitoring was essential to enable large-scale swarm experiments without human intervention.

Using this algorithmic approach, we conducted several large-scale experiments (Fig. 3 and movies S1 to S4). We programmed the full swarm of 1024 robots to self-assemble into two shapes, each taking roughly 12 hours of execution time. We performed 11 additional experiments with smaller collectives to analyze the accuracy of shape formation and to investigate the consistency of the collective behavior (19). All experimental trials fully assembled the desired shape without human intervention (such as charging batteries or reprogramming); hence, the collective behavior is remarkably robust. To quantitatively measure the accuracy of shape formation, we measured the mean square error between the

true position of each robot (as measured from a calibrated camera) and each robot's internal localized position in the shared coordinate system (Fig. 3G). We observed that the final shape accuracy is high but not perfect; errors by individual robots during the assembly process can translate into global warping of the shape or small packing defects, but without halting the self-assembly process. To test consistency, we used approximately 100 robots to form the same rectangular shape 10 times. The results show that although the shape accuracy values are comparable between experiments, the packing pattern of robots can exhibit considerable variability, as is true in natural self-assemblages. These patterns are a manifestation of the natural variation in agent behavior in such large groups.

Experiments also revealed several interesting emergent behaviors that occur during the formation process (Fig. 3). For example, "traffic jams" of edge-following robots formed behind particularly slow robots, similar to vehicle traffic. Another observed behavior was the complex "erosion" of the initial group, caused by edge-following robots that would accidentally push stationary

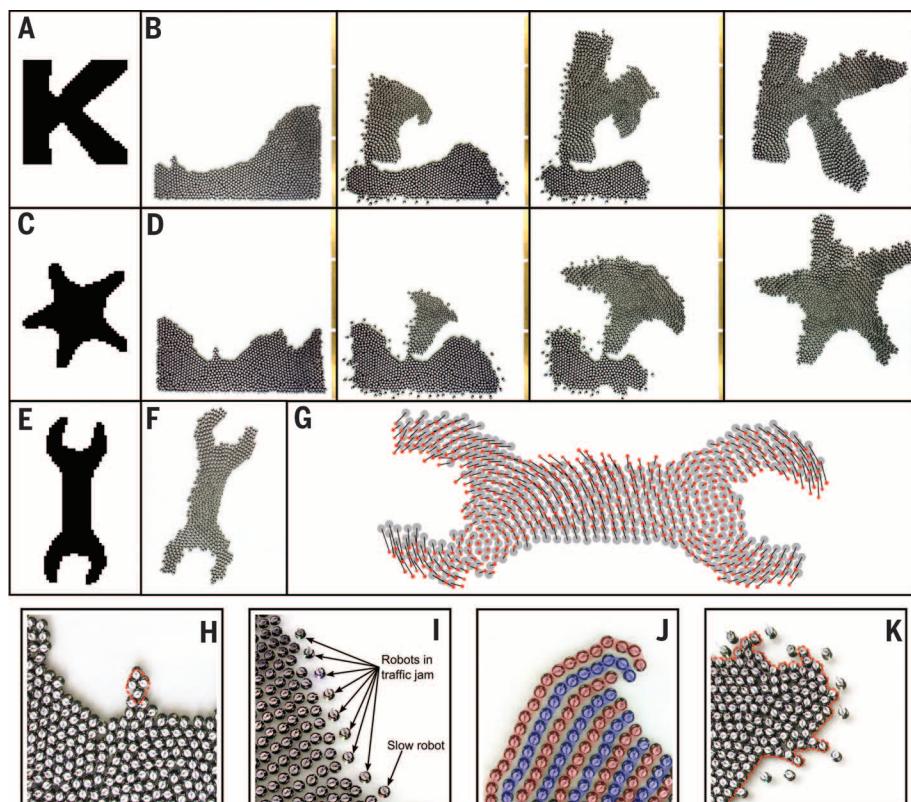


Fig. 3. Self-assembly experiments using up to 1024 physical robots. (A, C, and E) Desired shape provided to robots as part of their program. (B and D) Self-assembly from initial starting positions of robots (left) to final self-assembled shape (right). Robots are capable of forming any simply connected shape, subject to a few constraints to allow edge-following (19). (F) Completed assembly showing global warping of the shape due to individual robot errors. (G) Accuracy of shape formation is measured by comparing the true positions of each robot (red) and each robot's internal localized position (gray). (H to K) Close-up images of starting seed robots (H), traffic backup due to a slowly moving robot (I), banded patterns of robots with equal gradient values after joining the shape (J), and a complex boundary formed in the initial group (dashed red line) due to erosion caused by imprecise edge-following (K).

robots and create sharp edges; this resulted in a positive-feedback process where the sharp edges attracted more collisions, resulting in complex boundaries being formed as the initial group dissolved. Idealized mathematical models of robot swarms do not predict these phenomena.

Collective behavior in nature often involves large numbers of independent individuals interacting to produce complex assemblies. Engineered systems such as DNA-based self-assembly (27) have replicated this ability in synthesized chemical systems. We have demonstrated this ability in a robotic system by creating and programming a large-scale autonomous swarm to achieve complex global behavior from the cooperation of many limited and noisy individuals. The large-scale experiments advance our ability to engineer complex robotic systems. This motivates new investigations into advanced collective algorithms capable of detecting malfunctioning robots and recovering from large-scale external damages, as well as new robot designs that, like army ants, can physically attach to each other to form stable self-assemblies.

REFERENCES AND NOTES

- G. M. Whitesides, B. Grzybowski, *Science* **295**, 2418–2421 (2002).
- H. C. Berg, *Annu. Rev. Biochem.* **72**, 19–54 (2003).
- P. Lawrence, *The Making of a Fly: The Genetics of Animal Design* (Blackwell, Oxford, 1992).
- S. Camazine *et al.*, *Self-Organization in Biological Systems* (Princeton Univ. Press, Princeton, NJ, 2003).
- C. Anderson, G. Theraulaz, J. Deneubourg, *Insects Soc.* **49**, 99–110 (2002).
- N. J. Mlot, C. A. Tovey, D. L. Hu, *Proc. Natl. Acad. Sci. U.S.A.* **108**, 7669–7673 (2011).
- M. Kirschner, J. Gerhart, *The Plausibility of Life: Resolving Darwin's Dilemma* (Yale Univ. Press, New Haven, CT, 2005).
- R. Wood, R. Nagpal, G. Y. Wei, *Sci. Am.* **308**, 60–65 (March 2013).
- R. Grob, M. Bonani, F. Mondada, M. Dorigo, *IEEE Trans. Robot.* **22**, 1115–1130 (2006).
- E. William, G. Mermod, A. Martinoli, in *Proceedings of the 2010 International Conference on Robotics and Automation* (Anchorage, AK, May 2010), pp. 1438–1445.
- M. Yim *et al.*, *Robot. Automat. Mag.* **14**, 43–52 (2007).
- S. Goldstein, J. Campbell, T. Mowry, *Computer* **38**, 99–101 (2005).
- M. Tolley, H. Lipson, *Int. J. Robot. Res.* **30**, 1566–1584 (2011).
- P. Wurman, R. D'Andrea, M. Mountz, *AI Mag.* **29**, 9–19 (2008).
- K. Konolige *et al.*, *Exp. Robot.* **IX**, 131–140 (2006).
- J. McLurkin *et al.*, in *AAAI Spring Symposium: To Boldly Go Where No Human-Robot Team Has Gone Before* (Stanford, CA, March 2006), pp. 72–75.
- M. Rubenstein, C. Ahler, R. Nagpal, in *Proceedings of the 2012 IEEE International Conference on Robotics and Automation* (St. Paul, MN, May 2012), pp. 3293–3298.
- P. Vartholomeos, E. Papadopoulos, in *Proceedings of the 2006 IEEE International Conference on Robotics and Automation* (Orlando, FL, May 2006), pp. 649–654.
- See supplementary materials on Science Online.
- D. Moore, J. Leonard, D. Rus, S. Teller, in *The 2nd International Conference on Embedded Networked Sensor Systems* (Baltimore, November 2004), pp. 50–61.
- J. Werfel, Y. Bar-Yam, D. Rus, R. Nagpal, in *Proceedings of the 2006 IEEE International Conference on Robotics and Automation* (Orlando, FL, May 2006), pp. 2787–2794.
- D. Arbuckle, A. Requicha, *Auton. Robots* **28**, 197–211 (2010).
- K. Støy, *Robot. Auton. Syst.* **54**, 135–141 (2006).
- K. Tomita, S. Murata, H. Kurokawa, E. Yoshida, S. Kokaji, *IEEE Trans. Robot. Autom.* **15**, 1035–1045 (1999).
- M. De Rosa, S. Goldstein, P. Lee, J. Campbell, P. Pillai, in *Proceedings of the 2006 IEEE International Conference on*
- on Robotics and Automation* (Orlando, FL, May 2006), pp. 1462–1468.
- R. Nagpal, H. Shrobe, J. Bachrach, in *Information Processing in Sensor Networks*, F. Zhao, L. Guibas, Eds. (Springer, Berlin, 2003), pp. 333–348.
- P. W. Rothemund, *Nature* **440**, 297–302 (2006).

ACKNOWLEDGMENTS

Supported by the Wyss Institute for Biologically Inspired Engineering and by NSF grants CCF-0926148 and CCF-0643898.

SUPPLEMENTARY MATERIALS

www.sciencemag.org/content/345/6198/suppl/DC1
Materials and Methods
Figs. S1 to S10
Tables S1 and S2
Movies S1 to S4
References (28, 29)

3 April 2014; accepted 18 June 2014
10.1126/science.1254295

RNA NANOSTRUCTURES

A single-stranded architecture for cotranscriptional folding of RNA nanostructures

Cody Geary,¹ Paul W. K. Rothemund,² Ebbe S. Andersen^{1*}

Artificial DNA and RNA structures have been used as scaffolds for a variety of nanoscale devices. In comparison to DNA structures, RNA structures have been limited in size, but they also have advantages: RNA can fold during transcription and thus can be genetically encoded and expressed in cells. We introduce an architecture for designing artificial RNA structures that fold from a single strand, in which arrays of antiparallel RNA helices are precisely organized by RNA tertiary motifs and a new type of crossover pattern. We constructed RNA tiles that assemble into hexagonal lattices and demonstrated that lattices can be made by annealing and/or cotranscriptional folding. Tiles can be scaled up to 660 nucleotides in length, reaching a size comparable to that of large natural ribozymes.

Design methods for DNA (*1*, *2*) and RNA (*3*–*5*) nanostructures have been developing along parallel lines for more than a decade. Both allow the generation of a wide variety of structures, including polygons (*6*–*8*), three-dimensional (3D) shapes (*9*–*13*), 1D arrays (*3*, *14*), and 2D lattices (*15*, *16*). RNA structure design lags behind only in the size of structures created: Whereas RNA designs have been limited to 50 to 200 nucleotides (nt), DNA structures up to 45,000 nt have been achieved (*17*, *18*). The advantage of RNA structures lies in their greater potential to be cloned, genetically expressed in large quantities, and used for synthetic biology applications *in vivo*. DNA nanostructures built from single strands (*19*), or mostly single strands (*9*), have been synthesized or even cloned (*20*), but they variously require downstream purification or heat annealing, and none has yet been demonstrated to be folded isothermally as it is enzymatically produced.

RNA nanostructures have already been cotranscriptionally assembled from a mixture of numerous strands (*12*) and expressed in bacterial cells (*21*). Even if genetically encoded DNA structures are eventually achieved, other breakthroughs will have to be made, because existing DNA nanostruc-

tures lack the structural resolution and diversity of RNA tertiary motifs (*22*), such as kissing-loop interactions (KLs) (*23*), and they are currently unable to match the functional capacity of RNA to perform tasks such as scaffolding proteins (*21*, *24*), detecting ligands (*25*), or releasing small interfering RNAs in response to stimuli (*26*). But despite the potential for expressible RNA structures, a general architecture for designing them has yet to be achieved.

We present a method for designing single-stranded RNA (ssRNA) structures that can fold either by heat-annealing over mica or by cotranscriptional folding. Similar to the previously reported scaffolded DNA origami method (*17*), our ssRNA origami architecture is scalable and follows systematic design rules, in this case based on 3D modeling of the A-form helix. We demonstrated constructions up to six helices tall and 660 nt in length. We took inspiration from multi-stranded DNA double-crossover (DX) molecules (*27*) and their use as modular lattice-forming tiles (*16*). To turn multi-stranded DNA tiles into ssRNA ones, we developed an architecture that uses hairpins and programmable kissing loops. To extend the architecture to more than two helices, we introduced a new crossover pattern that we call a “dovetail seam,” for creating arrays of coaxially stacked helices that are both sequence-specific and nontopologically linked.

Our design method used 3D modeling of RNA (Fig. 1A) to plan the interconnection of helices and RNA structural motifs to form compact structures and lattices (Fig. 1B). Outputs of 3D

¹Center for DNA Nanotechnology, Interdisciplinary Nanoscience Center, and Department of Molecular Biology and Genetics, Aarhus University, 8000 Aarhus, Denmark.

²Bioengineering, Computer Science, and Computation and Neural Systems, California Institute of Technology, Pasadena, CA 91125, USA.

*Corresponding author. E-mail: esa@inano.au.dk