Programmable 3D Stochastic Fluidic Assembly of cm-scale Modules

Michael T. Tolley, Hod Lipson, *Member, IEEE*

***Abstract*—Self-reconfiguring modular robotic systems offer a potential route to achieving *programmable matter*, i.e. a substance the shape and properties of which can be tuned as required to achieve a variety of tasks. However, most modular robotic system designs rely on deterministic module motions which place significant power, control, and actuation requirements on the individual modules. This leads to relatively large modules and low target structure resolution. Here we experimentally demonstrate an alternative approach based on stochastic assembly, in which modules assemble into target structures in a fluidic tank. This system employs ambient fluid motion for module transportation. Assembly is directed by controlling the fluid flow through an active assembly substrate with an array of valves. Different valving programs are used with feedback from pressure sensors to achieve the completely automated hierarchical assembly of non-planar 3D structures.**

I. Introduction

Recently, there has been a great deal of interest in the concept of *programmable matter*, with a number of research groups pursuing a variety of approaches to achieving a robotic system with programmable morphology [1]-[15]. One potential route to achieving programmable matter is to employ stochastic assembly [6]-[15]. Inspired by nature, stochastic approaches achieve the assembly of robotic modules by taking advantage of stochastic environmental motions for component transportation. This reduces the power, actuation, and computation requirements on the modules, allowing them to be produced more cheaply, and at smaller scales, thereby increasing the resolution of the assembled structures.

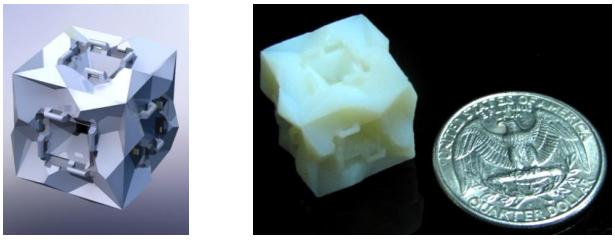
Stochastic Fluidic Assembly (SFA) represents a promising approach to achieving target geometries for programmable matter [8]-[13]. Previous SFA work has achieved the assembly of a small number of 10 cm to 20 cm scale modules [8],[9], and planar assembly of 500 micron scale modules [10],[11]. Computational studies have also explored control approaches for the assembly of large numbers of SFA modules [12],[13].

In this paper, we present results demonstrating the SFA of simple, completely passive, cm scale modules. In previous work, we presented an experimental system for cm-scale SFA and initial assembly experiments [14]. We build on these results here with an upgraded experimental system to demonstrate three advances that represent important milestones on the path to achieving 3D programmable matter through SFA: robust assembly, hierarchical assembly, and completely automated assembly. To our knowledge, this is the first time any of these have been achieved experimentally at this scale.

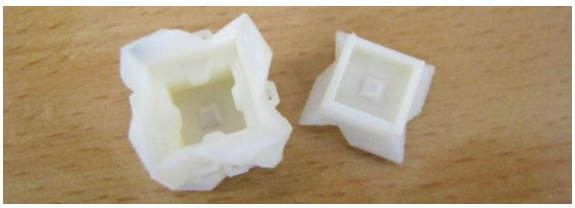
The remainder of this paper discusses these experimental results and the experimental system used to achieve them. In Section II we review the design of the experimental system and control software. Section III provides an overview of the control approach for the experiments. Section IV then presents the first set of experiments that demonstrate the robust assembly of non-planar 3D structures with minimal human intervention. In Section V we describe the second set of experiments demonstrating the parallel hierarchical assembly of 3D structures. We then describe a third set of experiments in which sensor feedback is used to achieve the fully automated assembly of 3D structures in Section VI. Finally Section VII comprises an overall discussion and conclusion of these results.

II. System Design

The design of our SFA system is discussed in detail in [14]. However, significant modifications were made to this experimental apparatus to enable the experiments described in this paper. In this section, we summarize the overall design of the SFA system, focussing on the new aspects.



(a) (b)



(c)

Fig. 1. SFA Modules. (a) Computer aided design and (b) 3D printed version of assembly module with a length of 15 mm. Alignment patterns and latches on each face allow cubes to self-align and bond together under the influence of fluidic assembly forces. (c) A two-piece design traps an internal air pocket to achieve neutral-buoyancy.

This work was funded by the DARPA grant number W911NF-08-1-0140 (Programmable Matter Program). M. T. Tolley would also like to thank the Canadian NSERC for their support through the PGS Program.

M. T. Tolley is with the Harvard University Microrobotics Laboratory, Cambridge, MA, 02138 USA (e-mail: mike.tolley@wyss.harvard.edu). H. Lipson is with the Computational Synthesis Laboratory, Cornell University, Ithaca, NY, 14853 USA (e-mail: hod.lipson@cornell.edu).

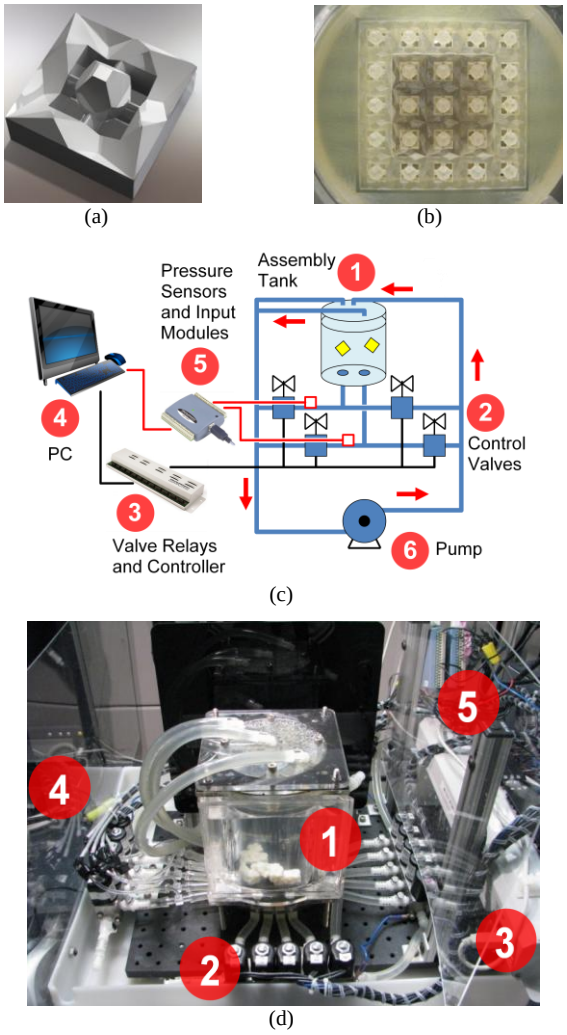


Fig. 2. Experimental system. (a) CAD drawing of outlet port design. (b) Photograph of assembly substrate with nine active ports surrounded by 16 passive ports. (c) Schematic, and (d) photograph of experimental system with primary components labeled.

A. *Stochastic Fluidic Assembly System*

The SFA modules are assembled on an active assembly substrate consisting of an array of fluidic ports with alignment patterns that match those of the cubes [Fig. 2(a)]. The substrate consists of a three-by-three array of active ports surrounded by 16 passive ports [Fig. 2(b)]. A schematic of the system that controls the active ports so that they act as either sources or sinks is seen in Fig. 2(c). A pump provides high pressure flow to a set of control valves that can be opened to make the active ports serve as fluid sources. A second, complimentary set of solenoid valves can be opened to connect the ports to the low pressure end of the fluidic circuit, causing them to serve as sinks. An alternate path allows fluid to flow through the top of the tank when all of the valves are closed. The solenoids are controlled with a USB relay controller. A set of pressure sensors provide feedback on the state of the active assembly ports (PX40-50BHG5V, Omega Engineering). This information is relayed to the control software with an A/D input modules (USB-1208FS, Measurement computing). Part numbers and suppliers for all the components not specified above can be found in [14]. Fig. 2(d) is a photograph of the experimental system.

B. *Control Software*

Custom control software was written in C++ to allow both manual and automated control of the SFA system. ON-OFF valve control was found to be too crude to control assembly (e.g. activating a source adjacent to a sink would completely reject an attracted module). In order to achieve a more continuous range of control options, we implemented pulse-width modulation (PWM) of the control signals to the valves. We tested the PWM valve control by connecting a flow meter to the outlet of a valve and measuring the resulting flow rates for various duty cycle settings at three different switching frequencies (Fig. 3). Ideally, the results of Fig. 3 could be used to improve the linearity of the flow rates. However, we had difficulty actuating the valves at higher than 20 Hz and the nearly linear results were found to provide a much more continuous range of fluid control. Additionally, the pulsated flow caused by the PWM switching was found to be beneficial to assembly as it helped to overcome friction/stiction between the cubes.

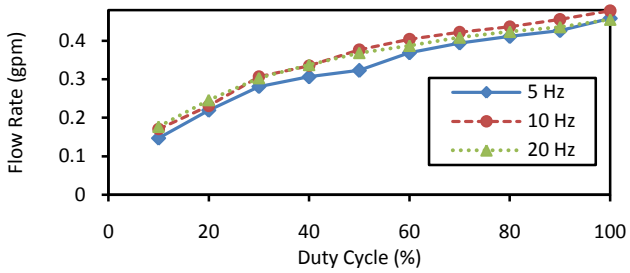


Fig. 3. PWM flow control. Flow rates measured for PWM at various frequencies confirm this as a valid approach to achieving flow control between completely *off* or *on*.

III. Control Algorithms

The custom control software described in the previous section also includes an implementation of a higher level control scheme used to automate SFA. This section describes the control approach, which has the goal of minimizing the feedback required for assembly. This is achieved by defining a set of open-loop algorithms to accomplish the various assembly tasks. The approach can be seen as a finite state machine (FSM) where feedback is only required to initiate the state transitions (Fig. 4). Each state represents a set of repeating valve sequences designed to achieve a particular assembly task, i.e. attraction, alignment, latching, or release. The initial inputs (e.g. the cube attraction locations) are defined by the user in the assembly program.

For example, a particular program may initiate assembly by identifying two attraction locations on the substrate, at which point control transitions to the *Attract* state. The identified locations become sinks while the rest of the substrate ports become sources. In order to agitate the cubes, the

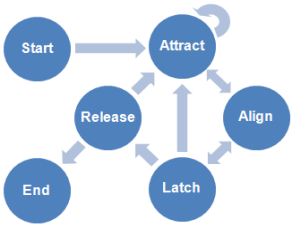


Fig. 4. Schematic of FSM approach to assembly control. Assembly consists of a sequence of four fundamental assembly operations corresponding to the four main states seen here. Each of the states runs open-loop valving sequences. Transitions between states are initiated in response to feedback on the state of the system according to the specific assembly program.

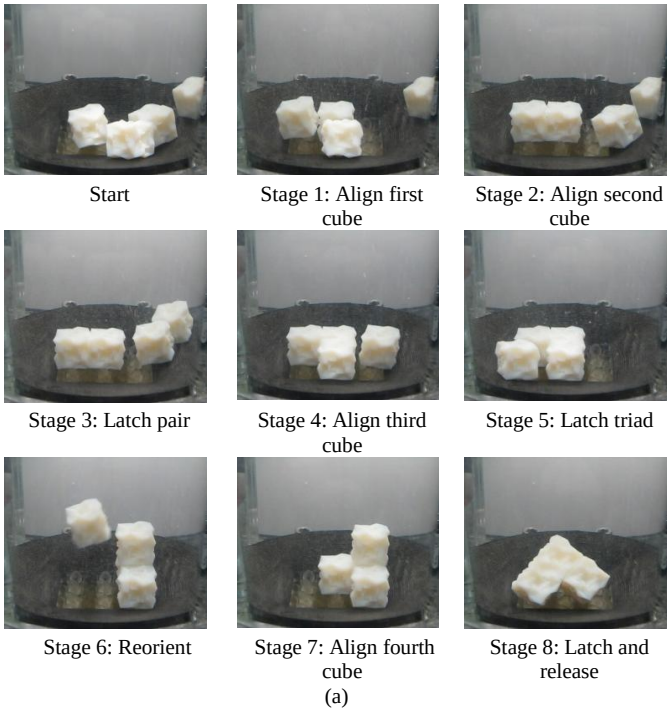
TABLE I
Finite-State-Machine Open-Loop Assembly Control Algorithms

State	Inputs	Algorithm
Attract	Desired cube locations	• Open desired cube locations to low pressure • Open all other ports to high pressure • Modulate duty cycle of high pressure Loop through sources one at a time, increasing their strengths to A_{max}, then back down to A_{min} (for agitation) • When one or more cubes have been attracted to desired locations, transition to Align
Align	Locations of mated and unmated cubes	• Open sinks at cube locations • Also open source at mated cube location and increase its strength slowly until cubes align • Done when cubes are aligned (go to latch), or a target location is no longer occupied by a cube (go to attract)
Latch	Cube locations	• Open sinks at cube locations • Open sources at cube locations • Increase source strengths slowly to vibrate cubes until cubes latch (go to attract or release as dictated by assembly plan), or become misaligned (go to align)
Release	Cube locations	• Open sources at cube locations • Slowly increase sources to release structure from substrate • For reorientation, attract to new location (attract), otherwise assembly is done
Align/ Latch	Location(s) of mated cube(s)	• Open sinks at desired cube locations • Agitate flow with remaining ports, as in Attract • Loop through sinks with attracted cubes, using PWM to switch them between sink and source to agitate the attracted cube and allow new cubes to align in adjacent locations • Continue to agitate attracted cubes until latched • Done when all cubes are aligned (go to latch), or a target location is no longer occupied by a cube (go to attract)

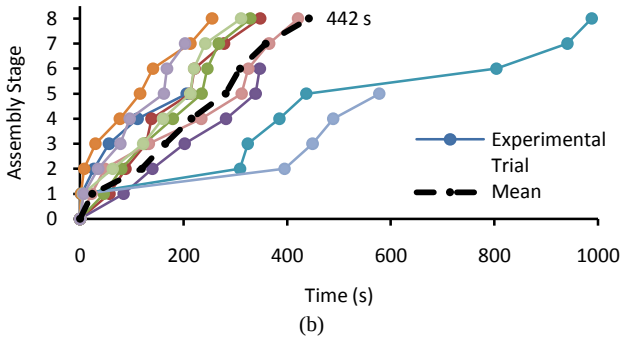
sources cycle between levels of high and low flow (using PWM). Once cubes have been attracted to the desired locations, the software moves to the *Align* stage. (Cube attraction is identified by operator feedback in the experiments of Sections IV and V, and sensor feedback in Section VI). Table I details the four assembly state algorithms, along with the required inputs. A fifth state *Align/Latch* is also listed that was developed to replace the separate *Align* and *Latch* algorithms for the automated experiments of Section VI.

IV. Robust 3-D Assembly

One of the key aspects of our SFA system is that the forces used to manipulate modules can also be used to manipulate assemblies. Because of this, a non-planar 3D structure can be assembled by releasing a 2D structure assembled on the substrate and re-orienting it into an upright position. New modules can then be added at the substrate to form 3D structure. To demonstrate this capability, we performed a set of experiments to assemble an L-shape, then reorient it upright on the substrate to add a fourth module [Fig. 5(a)]. In order to evaluate the robustness of this approach, we repeated this process 10 times consecutively and recorded the elapsed times for each assembly stage [Fig. 5(b)].



(a)



(b)

Fig. 5. Robust 3-D Assembly. (a) Sequence of images from a robust 3-D SFA experiment at eight stages of assembly. (b) Plot of assembly times of 10 consecutive 3-D assembly experiments. Each data series represents one experiment. Data points indicate the time elapsed since the start of the experiment. The dashed line indicates the mean time for each assembly stage.

Fig. 5(a) shows a sequence of images that illustrate the 8 stages required for the assembly of the four-module target structure. (Note that some of these stages comprise multiple of the FSM states discussed in the previous section). This sequence of stages represents the overall program for the assembly of the structure shown in the last image. For these experiments, the transitions between the FSM states required were initiated by a human observer.

Fig. 5(b) summarizes the time taken in each experiment to achieve the eight assembly stages depicted in Fig. 5(a). Each dataset represents one experiment, with the vertical axis representing the ordered assembly stages and the horizontal axis denoting the corresponding experiment time. Some of the experiments did not achieve all of the assembly stages. This is because an experiment was terminated if a structure inadvertently disassembled such that the entire program would have to be restarted from Stage 1.

Overall, the target structure was completed in 60% of the experiments. The mean time to the final assembly stage was 442 s whereas the median was 339 s (because the mean is largely affected by one outlier). The assembly times for each stage were fairly consistent with the exception of occasional delays at stages 2 and 6. Comparing these results with the experiments described in [14], we see the upgraded experimental system and improved control strategies reduced the average pair assembly time from 346s to 165 s, and the three-module assembly time from 398 s to 281 s.

V. Parallel Hierarchical 3D Assembly

One of the challenges of SFA highlighted by the results of the previous section is the slow assembly rates relative to deterministic assembly approaches. Parallel hierarchical assembly provides a potential route to mitigating this disadvantage. This approach assembles multiple sub-components in parallel to later be assembled into a final structure, significantly reducing the overall assembly time (especially if the goal is to manufacture batches of identical structures).

In order to demonstrate our system's ability to achieve parallel hierarchical assembly, we conducted experiments to assemble structures that form the building blocks for subsequent assembly. In these experiments, sub-structures composed of two modules were assembled, which were then assembled into two identical non-planar four-module structures. These experiments were conducted using a similar approach to that described in the previous section, using a FSM control algorithm with feedback from an operator.

Fig. 6 presents frames from videos of the two stages of the parallel hierarchical stochastic fluidic assembly of two 3-D structures. Non-planar structures were achieved by attracting the pre-assembled pairs to the substrate in both horizontal and vertical positions, and latching them together. Fig. 6 shows both the addition of an upright pair to a horizontal one, and the addition of a horizontal pair to an upright one. Parallel hierarchical assembly was achieved starting with enough modules for the assembly of two 3D structures (i.e. eight modules). Pairs of modules were assembled simultaneously on the top and bottom rows of the substrate. Because of the limited array size, however, the second stage of hierarchical assembly had to be conducted serially. Nonetheless, these experiments demonstrated the capability of the SFA system to achieve parallel hierarchical 3D assembly.

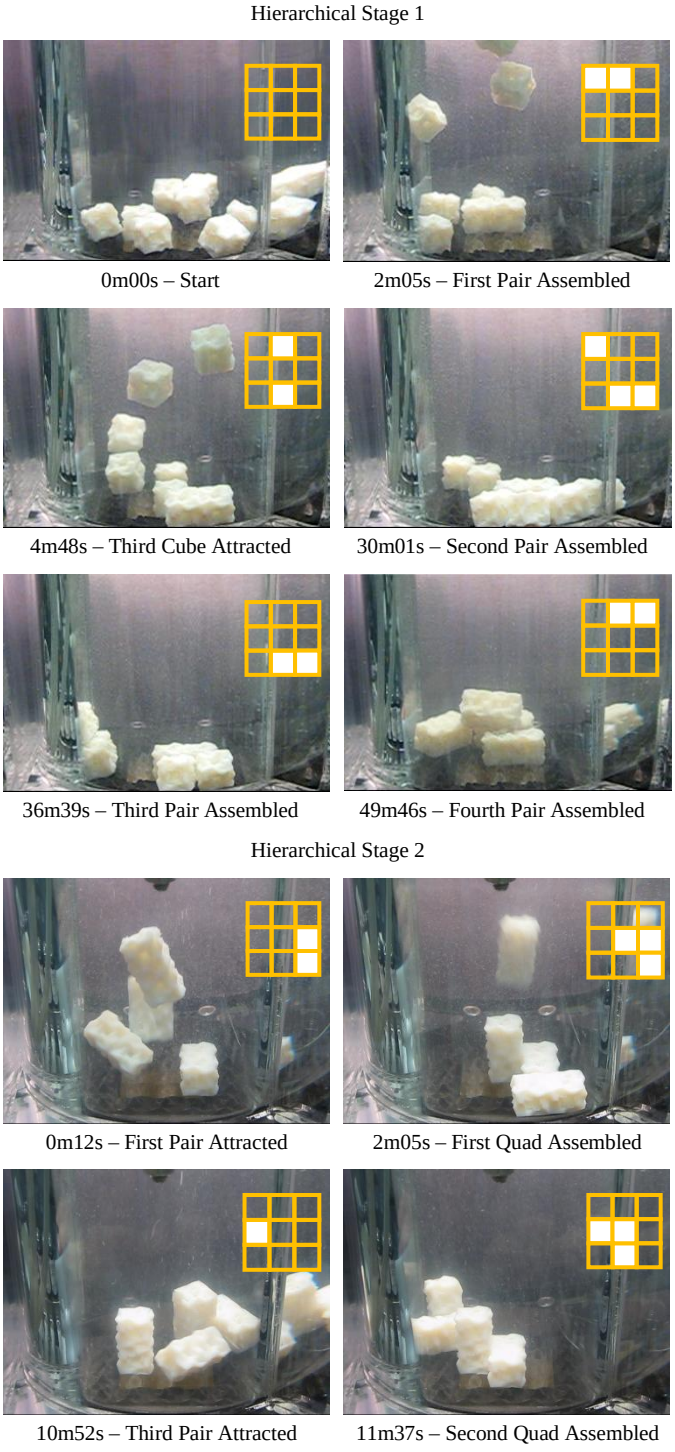


Fig. 6. Parallel Hierarchical 3D Assembly. Representative frames from two assembly experiments corresponding to the first stage and second stages of the parallel hierarchical assembly of two 3D structures. The grid on the upper-right hand corner of each image indicates the locations on the 3 × 3 array of active substrate ports that are occupied by modules.

VI. Automated Assembly Experiments

The goal of the third set of experiments was to demonstrate completely automated assembly of non-planer 3D structures. This was achieved following a hierarchical approach as in the experiments of Section V, except with pressure sensor feedback in the place of feedback from a human operator (see supplementary movie). Pressure sensors were used to determine when a module had been attracted to a substrate port. A module blocking an open sink caused relatively large decrease in the pressure in the outlet. Calibration experiments were used to tune the detection threshold such that a majority of cube attractions were detected while minimizing false positives. In order to detect the presence of cubes at the sinks, an assembly program's PWM switching was periodically suspended to allow continuous flow during sensor readings. The measured pressures were then compared with those of the previous reading. Each trial began with an initial reading of the unblocked flow pressures to account for changes in the overall pressure of the system.

Using this approach, we found that complete mating of the cube with the substrate was required to cause a detectable drop in the measured pressure. This is in contrast to previous visual detection method in which it was possible to identify when cubes were attracted to an outlet but not completely aligned. For this reason, for the automated system, the open-loop subroutines for *Attract* and *Align* had to be combined into one routine that could accomplish both (see Table I). This resulted in slower assembly rates since attracted modules would often be inadvertently rejected from the substrate. However, because of the precise mating required for automatic detection, the pressure sensors were more accurate than a human observer at identifying when adjacent modules had latched.

One of the consequences of using only pressure sensor feedback for assembly was it became much more difficult to define assembly programs to achieve a specific target structure. However, this is a common property of many stochastic assembly systems. For this reason, some researchers have proposed an alternative view of stochastic assembly. Instead of defining an assembly plan as one would for a deterministic system, we must focus on identifying assembly paths similar to chemical reactions, the rates of which can be tuned to achieve bulk quantities of desired target structures [6]. This is the view taken in these automated experiments.

We tested three basic assembly programs, recorded all of the resulting assembly and disassembly paths and recorded the rates of each (Fig. 7). (Disassembly events occurred randomly when the separation forces applied to cubes by fluid agitation and collisions were greater than the latching forces holding the assemblies together.) These statistics were collected over multiple 30 to 40 minute runs of each program.

In Program 1, the assembly tank was populated with individual modules, which were attracted to two adjacent substrate locations as indicated in Fig. 7(a). As described in the *Attract/Align* algorithm in Table I, the remaining other substrate ports acted as agitation sources, and a single attracted cube was agitated mildly to enable the alignment of the second cube. Once cubes were detected at both of the adjacent

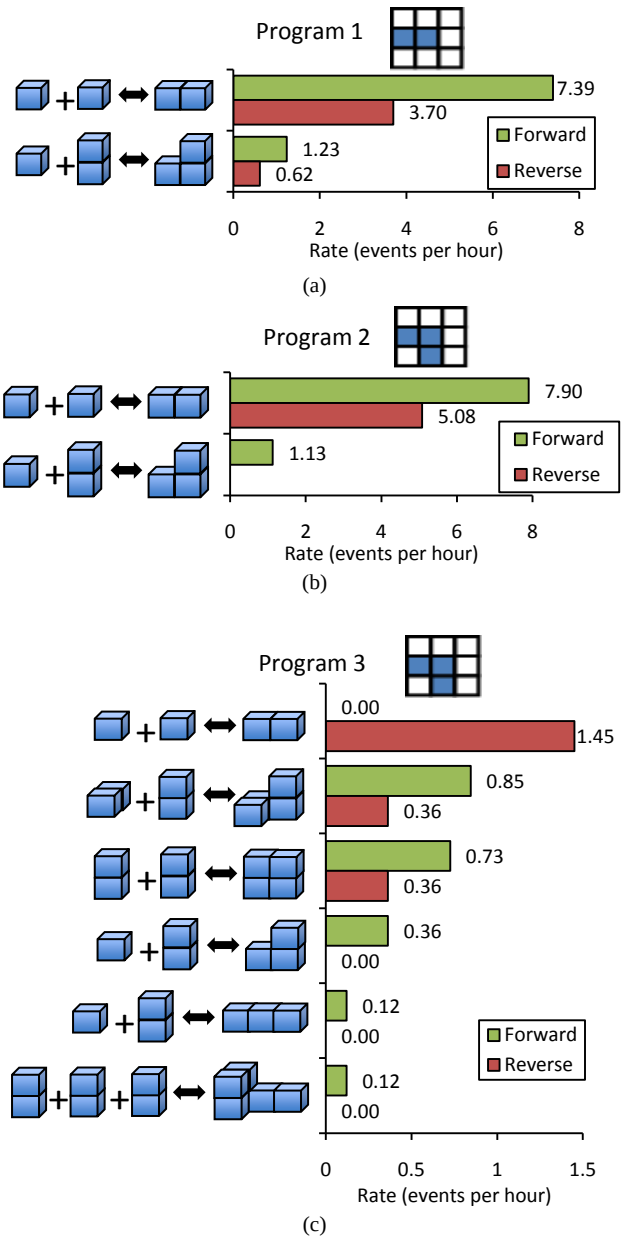


Fig. 7. Assembly rates. Bar graphs indicate the experimentally determined rates of various 'reactions' for three assembly programs. The light bars indicate the rates of the forward reaction (assembly), while the dark bars indicate reverse reaction rates (disassembly). Dark squares in the schematic to the top-right of each graph indicate the configuration of substrate ports acting as outlets in each case.

sinks, the assembled structure was released back into the flow and the program was repeated.

Program 2 was similar to Program 1 except with three substrate positions forming an L-shape acting as sinks [see Fig. 7(b)]. In Program 2, the tank was also initially populated with individual modules. Our hypothesis was that this program would lead to an increased production of three-cube L-shaped structures horizontally on the substrate. Program 3 involved the same outlet pattern as Program 2, but applied to a tank populated initially with cube pairs. The expectation was that this case would result in an increased production of 4-module assemblies, including non-planar structures.

Fig. 7 summarizes the results of the automated assembly experiments. Program 1 was run for a total of 1h 37m. Over this period, we observed the assembly of module pairs at a rate of 7.39 per hour, while the reverse reaction (disassembly of a pair into individual modules) occurred at a rate of 3.70 per hour. Thus, pairs were assembled at a net rate of 3.69 per hour. Occasionally, a pair would be attracted upright and assembled to a third cube to form a triad. This reaction occurred at a much lower net rate of 0.61 per hour. Thus, the primary result of Program 1 was to module pairs.

Program 2 was tested over the course of 1h 46m. One of the consequences of the *Attract/Align* subroutine for programs with three outlet ports is that often the agitation of previously assembled modules required to allow a third module to align often inadvertently reject the incomplete structure from the substrate. Thus, this program frequently resulted in the assembly of module pairs, and less frequently in L-shapes. Nonetheless, the net assembly rate of L-shapes (1.13 per hour) was almost twice that of Program 1, while that of pairs (2.82 per hour) was approximately one fourth.

Finally, Program 3 resulted in a range of reactions at a variety of reaction rates. Because of this, the program was run many more times than the others (a total of 8h 16m). Since the tank was repopulated with cube pairs for each run, the most common reaction was the disassembly of these pairs into individual modules. However, the second most common reaction was the assembly of pairs into the non-planar 3D structure assembled in the experiments of Section V.

VII. Discussion and Conclusions

In this paper we have presented a system for SFA and examined its capabilities with three sets of experiments. The first set of experiments demonstrated the ability of the system to achieve repeatable 3D assembly of four-module structures. This was achieved using a human operator to identify assembly events.

The second set of experiments demonstrated the parallel hierarchical assembly of two identical 3D structures. The assembly times of these experiments were relatively long due to the limited number of modules (excess modules hindered imaging and the visual identification of assembly), the operator feedback required to identify assembly events (which is inherently serial), and the limited size of the assembly substrate which prevented parallelization of the second assembly stage. Nonetheless, these experiments demonstrate the parallel hierarchical approach that may help overcome the slow assembly rates inherent with SFA.

In the third set of experiments, we used pressure sensor feedback to achieve completely automated 3D assembly. Three assembly programs were tested over the course of many hours and the resulting 'reactions' and associated reaction rates were recorded. These reactions resulted in different net assembly rates for different structures, suggesting that it may be possible to identify programs to achieve significant numbers of a desired target structure. However, many disassembly events occurred and their frequency seemed to increase over time. This suggests that the 3D printed latching mechanisms were deforming over time. Thus, improving the latch material and/or design could lead to more consistent results. Additionally, a relatively small number of a basic set of programs were tested. More experiments with a wider set of assembly programs would be necessary to draw conclusions about how to program the SFA of batches of more complex target structures.

Overall, this work demonstrates the potential of SFA as a route to achieving programmable matter. However, in order to program the matter's properties in addition to its shape, additional functionality will need to be embedded into the modules themselves (e.g. computation, sensors, actuators). Part of the advantage of the approach presented here is that none of this functionality is required to program the shape. This will allow the size of the modules to be decreased to improve the resolution of assembled structures. Complementary research is underway to address the issues of scaling and adding functionality to the SFA modules [11],[15].

References

[1] S. C. Goldstein, J. D. Campbell, and T. C. Mowry, "Programmable matter," *Computer*. vol. 38, iss. 6, pp 99-101, May 2005.

[2] K. Gilpin, K. Kotay, D. Rus, I. Vasilescu. "Miche: modular shape formation by self-disassembly," *Int. J. Robotics Research*, vol. 27, pp. 345-372, Mar. 2008.

[3] E. Hawkes, B. An, N. M. Benbernou, H. Tanaka, S. Kim, E. D. Demaine, D. Rus, and R. J. Wood. "Programmable matter by folding," *Proc. Natl. Acad. Sci.*, vol. 107, pp 12441-12445, July 2010.

[4] P. J. White, M. L. Posner and M. Yim, "Strength analysis of miniature folded right angle tetrahedron chain programmable matter," in *Proc. IEEE Int. Conf. Robotics and Automation,* Anchorage AK, May 2010, pp. 2785-2790.

[5] P. J. White, K. Kopanski, H. Lipson, "Stochastic self-reconfigurable cellular robotics," in *Proc. IEEE Int. Conf. Robot. and Autom.*, New Orleans LA, Apr. 2004, vol. 3, pp. 2888-2893.

[6] E. Klavins, S. Burden and N. Napp, "Optimal Rules for Programmed Stochastic Self-Assembly", in *Proc. Robotics: Science and Systems*, Philadelphia PA, Aug. 2006. pp. 9-16.

[7] S. Griffith, D. Goldwater, J. M. Jacobson, "Robotics: self-replication from random parts," *Nature* vol. 437, p. 636, Sept. 2005.

[8] P. White, V. Zykov, J. Bongard, and H. Lipson, "Three dimensional stochastic reconfiguration of modular robots," in *Proc. Robotics Science and Systems*, Cambridge, 2005, pp. 161-168.

[9] V. Zykov, and H. Lipson H., "Experiment design for stochastic three-dimensional reconfiguration of modular robots", in *Proc. IEEE Int. Conf. Intelligent Robots and Systems, Self-Reconfigurable Robotics Workshop*, San Diego CA, Oct. 2007.

[10] M. T. Tolley, M. Krishnan, D. Erickson, and H. Lipson, "Dynamically programmable fluidic assembly," *Appl. Phys. Lett.*, vol. 93, 254105, Dec. 2008.

[11] M. T. Tolley, A. Baisch, M. Krishnan, D. Erickson, and H. Lipson, "Interfacing methods for fluidically-assembled microcomponents", in *Proc. IEEE Int. Conf. Microelectromechanical Systems*, Tucson AZ, Jan. 2008, pp. 1073-1076.

[12] M. T. Tolley, M. Kalontarov, J. Neubert, D. Erickson, and H. Lipson, "Stochastic modular robotic systems: a study of fluidic assembly strategies," *IEEE T. Robotics*, vol. 26, pp. 1552-3098.

[13] M. T. Tolley and H. Lipson, "On-line assembly planning for stochastically reconfigurable systems," *Int. J. Robotics Research*, online first, Mar. 2011.

[14] M. T. Tolley and H. Lipson, "Fluidic Manipulation for Scalable Stochastic 3D Assembly of Modular Robots," in *Proc. IEEE Int. Conf. Robotics and Automation*, Anchorage AK, May 2010, pp. 2473-2478.

[15] J. Neubert, A. Cantwell, S. Constantin, M. Kalontarov, D. Erickson, H. Lipson, "A Robotic Module for Stochastic Fluidic Assembly of 3D Self-Reconfiguring Structures", *Proc. Int. Conf. on Robotics and Automation*, Anchorage AK, May 2010, pp 2479-2484.