

Two Computational Primitives for Algorithmic Self-Assembly: Copying and Counting

Robert D. Barish,[†] Paul W. K. Rothmund,[‡] and Erik Winfree^{*,‡,§}

Department of Computer Science, California Institute of Technology and Computation and Neural Systems, Pasadena, California 91125

Received October 14, 2005; Revised Manuscript Received October 29, 2005

ABSTRACT

Copying and counting are useful primitive operations for computation and construction. We have made DNA crystals that copy and crystals that count as they grow. For counting, 16 oligonucleotides assemble into four DNA Wang tiles that subsequently crystallize on a polymeric nucleating scaffold strand, arranging themselves in a binary counting pattern that could serve as a template for a molecular electronic demultiplexing circuit. Although the yield of counting crystals is low, and per-tile error rates in such crystals is roughly 10%, this work demonstrates the potential of algorithmic self-assembly to create complex nanoscale patterns of technological interest. A subset of the tiles for counting form information-bearing DNA tubes that copy bit strings from layer to layer along their length.

The challenge of engineering complex devices at the nanometer scale has been approached from two radically different directions. In top-down synthesis, information about the desired structure is imposed by an external apparatus, as in photolithography. In bottom-up synthesis, structure arises spontaneously due to chemical and physical forces intrinsic to the molecular components themselves. A significant challenge for bottom-up techniques is how to design molecules containing information capable of guiding the self-assembly process to create the desired supramolecular structure. Borrowing a trick from biology, information can be stored in a heteropolymer that will subsequently fold and/or assemble with other molecules as directed by the sequence information. This approach has been especially fruitful for designing nanostructures using the two premier biological polymers, peptides^{1,2} and nucleic acids.^{3,4} Biological organisms must use additional principles to create large structures with sophisticated organization, such as whales, containing upward of 10²⁴ macromolecules. To make such structures, information cannot be used as a blueprint that directly specifies where each molecule or cell should go (there is not enough information); it must instead specify the target structure implicitly via a set of rules (an algorithm) for how to construct it according to growth processes. This general principle, that computation has an essential role in construction tasks, was clearly expounded by von Neumann in his

study of self-reproducing machines.⁵ The complex genetic regulatory circuits that direct development are perhaps the most conspicuous example of this principle.

Somewhat surprisingly, computation also arises naturally within phenomena used by existing bottom-up fabrication methods, such as macromolecular self-assembly.^{6,7} Building on Wang's insights connecting geometry and computation in the context of mathematical tiling theory,^{8,9} *algorithmic self-assembly* has been proposed as a generalization of crystal growth processes and is capable of creating complex shapes and patterns.^{10–14} In this paradigm, assembly instructions are programmed into a set of tiles by choice of specific binding interactions, and self-assembly occurs spontaneously. Simple examples of algorithmic self-assembly have been demonstrated, first in one-dimension^{15,16} and then in two dimensions,¹⁷ using molecular Wang tiles made of DNA.^{18,19} Here we show that algorithmic self-assembly can be used to create an extended pattern—binary counting—of technological relevance for molecular electronics as the layout for a demultiplexing circuit^{20–24} and of fundamental theoretical interest due to its appearance as a primitive for many other computation and construction tasks.^{25–27} Additionally, using a subset of these tiles we show that a string of binary information can be propagated along the length of a DNA tube, which is of independent interest to the study of crystal evolution and the origin of life.²⁸

The binary counter pattern consists of an array in which each row represents an integer in binary, and each subsequent row represents the integer following the one below it. Table 1 shows counting from 0 to 8. In principle, this array could

* To whom correspondence may be addressed. E-mail: winfree@dna.caltech.edu.

[†] California Institute of Technology.

[‡] Computation and Neural Systems, California Institute of Technology.

[§] Department of Computer Science, California Institute of Technology.

Table 1

	⋮	⋮	⋮	⋮
⋯	1	0	0	0
⋯	0	1	1	1
⋯	0	1	1	0
⋯	0	1	0	1
⋯	0	1	0	0
⋯	0	0	1	1
⋯	0	0	1	0
⋯	0	0	0	1
⋯	0	0	0	0

be extended to the left with zeros, and extended upward to make an unbounded counter. Without any a priori knowledge of integer arithmetic or binary representation, this table can be generated by following simple rules based on the logic of the classical ripple-carry adder.²⁹ Starting with a row of all zeros, complete each subsequent row as follows. Provide a “carry bit” to the rightmost position, instructing it to increment. Then evaluate each position, from right to left: Write a 1 if either a carry bit was received or if the bit directly below on the previous row was 1, but not both. Otherwise, write a 0. Additionally, if both a carry bit and a 1 are received, provide a carry bit to the next position to the left. This can be summarized in equations as

$$b_{i+1}^{n+1} = b_{i+1}^n \text{ XOR } c_i^{n+1}$$

$$c_{i+1}^{n+1} = b_{i+1}^n \text{ AND } c_i^{n+1}$$

where b_i^n is the bit in the i th column and n th row of the binary counting array, and c_i^n is the carry bit provided by that position to its leftward neighbor.

These rules for constructing the binary counting table can be translated into a set of geometrical tiles, known as Wang tiles, that represent the 0’s and 1’s by their color and enforce the logical rules by markings or indentations on their sides. This is possible because the rules for computing each bit are local: each bit depends only on the bit below and the carry bit from its right. As shown in Figure 1a, each tile contains four “binding domains” whose shape either matches information provided by the tile below and by the tile to its right (the inputs) or provides information to the tiles above and to its left (the outputs). There are four rule tiles, one for each possible combination of inputs; each rule tile corresponds to a single entry in the lookup table for the equations given above. Thus, at sites where a new tile can bind by both of its input domains (such as the top of Figure 1b), there is a unique tile that can do so. This ensures that assemblies without mismatches will produce the correct binary counting pattern, if started from a row of zeros.³¹ Note that if tiles may be added only when they bind by at least two domains, which we call *legal growth*, then a finite-sized assembly cannot grow beyond existing rows and columns; for example, the assembly shown in Figure 1b will add exactly five tiles on the upper right and four tiles at the lower left. (Although they bind using different sides of the tiles, the latter tile additions are also each uniquely determined and correspond to counting backward!).

Unlike previously considered tile sets for binary counters,^{25,32} for experimental simplicity the tile set and scaffold discussed here do not provide a mechanism for initiating the counter at a specific number. Presuming that growth begins with just a column of blue scaffold tiles of some finite length, there is in fact *no* site where legal growth can occur. However, when a tile attaches by a single binding domain for sufficiently long, termed an *insufficient attachment*,³³ subsequent legal growth will complete the column above and below the initial tile. This initial tile may, randomly, represent either a 0 or a 1. Effectively, this means that rather than starting at zero, our crystals will seem to start at a random number, but count correctly thereafter. If more than one insufficient attachment occurs in a given column, however, then it is possible that the random choices will be incompatible with each other, and an error in the pattern will result. It is therefore important that, when implemented molecularly, tiles attaching by a single binding domain will be rare under the physical conditions that we will study.

Abstract Wang tiles can be translated directly into artificial DNA molecules, such as double-crossover molecules,¹⁸ by encoding the shapes of the abstract tile’s four binding domains as four stickyend sequences on the corresponding molecular tile;³¹ complementary shapes correspond to complementary sequences. This results in the set of four DAE-E molecules shown in Figure 1c. (Although this double-crossover motif has been shown previously to have intrinsic curvature that encourages assemblies of tiles to roll up into tubes,^{17,34} assemblies grown from long scaffolds in that work usually contained 5 to 15 layers of rule tiles, which we predicted would be sufficient for our investigations here.) To distinguish tiles representing 0’s from tiles representing 1’s, we decorated the latter with protruding hairpin motifs that provide topographic contrast when imaged by atomic force microscopy (AFM). Creating long, covalently bound columns of scaffold tiles required the synthesis of a long scaffold strand with periodic sequence, as shown in Figure 1d. As in previous work using scaffold strands,^{17,35,36} the molecular implementation consists of a long single-stranded DNA molecule upon which short oligonucleotides assemble to form a molecular structure equivalent to several fully formed tiles covalently joined along one strand at each sticky-end. Unlike the previous work, in which the overall path of the scaffold strand could be routed such that it never forms intentional base pairs with itself, the scaffold strand for the binary counter design here must wrap from helix to helix along a diagonal, which requires hairpin sequences at periodic intervals. To reduce problematic interference of secondary structure during synthesis of the scaffold strand by assembly PCR,³⁷ we chose exclusively-AT sequences for the hairpin domains. Furthermore, to specifically amplify the single-stranded sense strand at the end of assembly PCR (which produces long periodic double-stranded DNA), the entire scaffold strand sequence consists of only A,T,C; the final stage of synthesis, therefore, is provided only those nucleotides. Experimental protocols and the sequences of all strands used here are given in the Supporting Information.

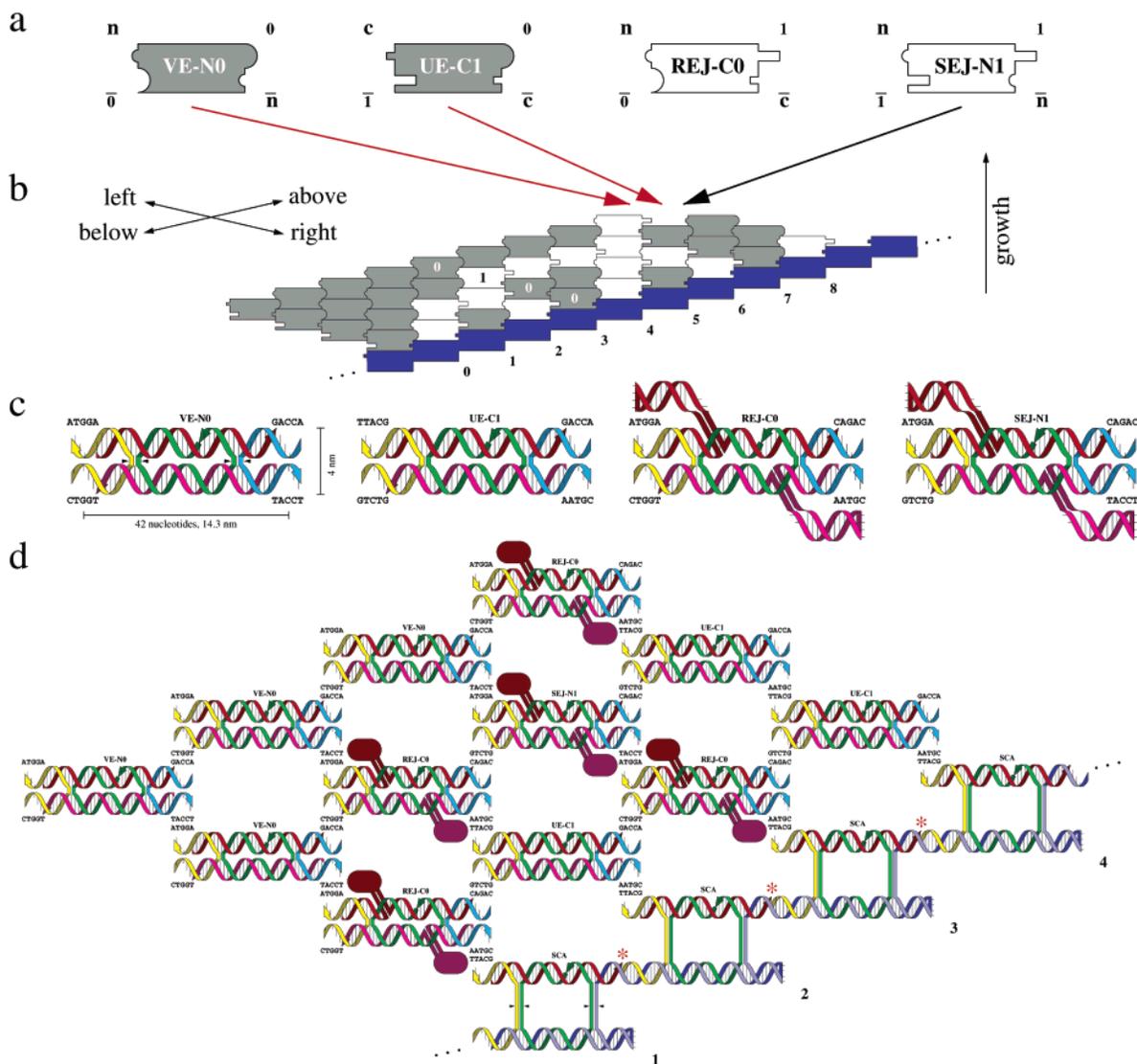


Figure 1. Abstract binary counter tile set and DNA Wang tile implementation. (a) The four binary counter rule tiles, VE-N0, UE-C1, REJ-C0, and SEJ-N1, corresponding to the four possible input pairs for ripple-carry adder logic. The first part of each tile name refers to the DNA tile core sequences used in previous work,³⁰ while the second part refers to the input pair that the tile matches. The two lower binding domains (with names containing overbars) on each tile act as inputs, while the upper two act as outputs. Each tile outputs either 0 or 1 to the tile above it and outputs either a carry bit (c) or not (n) to the tile to its left. (b) Assembly of the rule tiles on a linear scaffold (blue). At sites where a new tile can attach by both input binding domains, a unique tile matches correctly (black arrow) and two tiles match partially (red arrows). To compare to Table 1, see inset for orientation. The tiles in the row representing the number 4 (0100 in binary) have been explicitly labeled in the diagram. (c) Molecular implementation of the four rule tiles as DNA Wang tiles. Each tile is assembled from five single strands: two of 37 nucleotides (nt) (top & bottom, #1 & #5, red & magenta), two of 26 nt (left & right, #2 & #4, yellow & green), and one of 42 nt (central, #3, blue). For two tiles, hairpin-containing 59-mers replace the 37-mers, providing topographic contrast for AFM imaging. Triangles mark two crossover points, separated by two helical turns (21 nt). Arrowheads point from 50 to 30. Sticky ends (5 nt) are at the ends of the #2 and #4 strands, and have sequences corresponding to the logical labels in (a). (d) Self-assembly of the DNA Wang tiles on the scaffold (not to scale). Diagram is exploded to show matching of complementary sticky ends. Crossover points in the scaffold are stretched in the diagram to accommodate the exploded spacing, but the molecules contain no nucleotides at the crossover points; the secondary structure of the SCA scaffold tile is consistent with the DAE-E motif. The scaffold consists of a single long periodic scaffold strand (blue) and three scaffold tile strands (SCA; red, yellow, green, of lengths 37, 26, 42). The intrinsic curvature of the DAE-E tiles is such that the radius of curvature points up out of the page; red stars indicate diagram artifacts at the nicks due to flattening the structure.

To determine whether the designed tiles assemble as expected, we first tested the tiles without the scaffold. In this case, the intrinsic curvature of DAE-E tiles was expected to encourage the formation of tubes. In fact, the tiles here called VE-N0 and SEJ-N1 are identical to the tiles VE-00 and SEs(1,5:h14) reported in ref 34 to each form single-tile tubes, frequently longer than 5 μm with a circumference

typically between 4 and 12 tiles and with the tile axis parallel to the length of the tubes. In this work, single-tile tubes are prepared by mixing each relevant strand at 200 nM in TAE/Mg²⁺ buffer (40 mM Tris acetate, 1 mM EDTA, 12.5 mM Mg acetate, pH 8.3) and annealing from 90 to 20 °C at 1 °C/min. After 1–24 h at room temperature, samples were examined by adding 5 μL to a 30 μL droplet of annealing

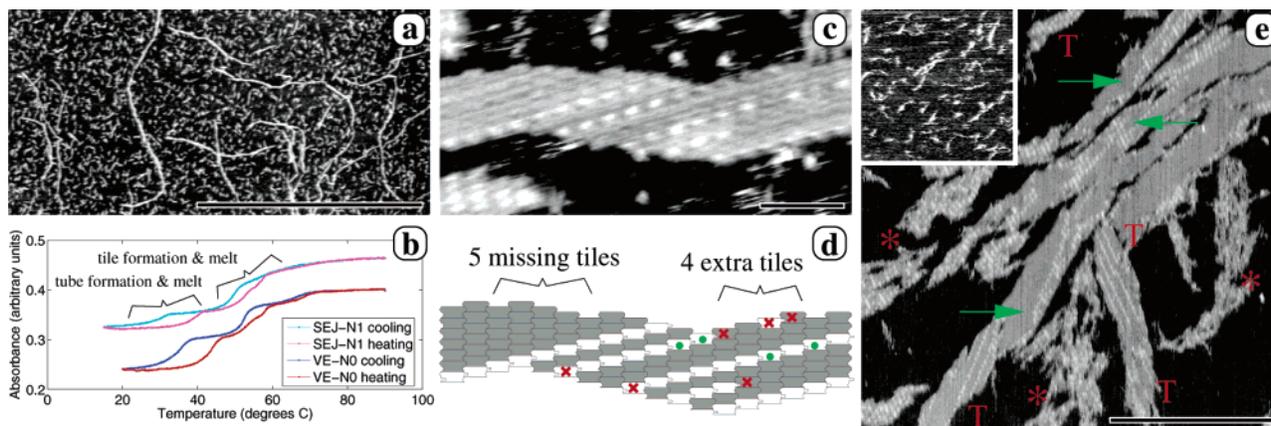


Figure 2. Tubes and crystals (a) Tiles VE-N0 and SEJ-N1 annealed together (at 200 μM each tile) form tubes several micrometers in length, which we term COPY tubes. AFM image. Scale bar is 5 μm . (b) Thermal formation and melting profiles of VE-N0 tubes (lower trace) and SEJ-N1 tubes (upper trace), as measured by hypochromicity at 260 nm. (c) Detail of an opened COPY tube. AFM image. Scale bar is 50 nm. (d) Interpretation of the COPY tube seen in (c). Red crosses mark tiles that mismatch their leftward neighbors. Green dots indicate tiles presumed to have lost their hairpins or that were poorly imaged by AFM. Note that the tube has split parallel to the tube axis. If the tube had a constant circumference of seven tiles, then five tiles must have fallen off as the tube opened, and four tiles subsequently attached after the tube had opened on the mica. (e) All four binary counter tiles annealed together with the scaffold strand. Green arrows indicate putative binary counting patterns growing from scaffold. Red “T”s mark what appear to be tubes nucleated without a scaffold strand. Red stars indicate ill-formed assemblies of undefined nature. The inset shows scaffold strand annealed with just UE-C1 and REJ-C0, which should assemble with just a single layer of tiles on the scaffold tiles. Typical lengths of the scaffold are mostly in the range 50–500 nm. Scale bar is 500 nm. Inset is the same scale.

buffer on freshly cleaved mica for AFM imaging using fluid tapping mode (as described for counting “tube falling events” in ref 34). During AFM imaging, after landing on the mica, tubes first appear as thick indistinct linear objects (closed tubes) and subsequently pop open to lie flat on the mica such that individual tiles can be resolved.

Assembly using both tiles VE-N0 and SEJ-N1 together (and no others) is a particularly interesting case: these being the tiles without any carry bit input or output, they copy a bit pattern unchanged from one layer to the next. Hence we call the resulting algorithmic assemblies “COPY tubes”. Like the single-tile tubes, COPY tubes can grow to multi-micrometer lengths (Figure 2a). The bit pattern carried by a particular tube is determined during the nucleation stage (which we assume to be random with some bias) and copied from layer to layer by growth processes thereafter. Because the axis of curvature is perpendicular to the helix axes of the tiles, the resulting stripe patterns should appear as diagonal stripes that wrap helically around the tubes. However, samples prepared with the single-tile tube annealing schedule were found to predominantly consist of either all-VE-N0 tubes or all-SEJ-N1 tubes (distinguished by the presence of their topographic label).

To investigate whether this disproportionation could be due to a difference in the binding affinities of the two tiles, we examined the thermal formation and melting profiles of each single-tile tube, using UV spectrophotometry. Degassed 1000 μL samples with each strand at 200 nM in the annealing buffer were heated from room temperature to 90 $^{\circ}\text{C}$ at 0.4 $^{\circ}\text{C}/\text{min}$ to fully melt the samples and equilibrate the instrument. Data were collected as the samples were then cooled at 0.15 $^{\circ}\text{C}/\text{min}$ to either 15 or 20 $^{\circ}\text{C}$, held for 2 h, and then heated back to 90 $^{\circ}\text{C}$ at 0.15 $^{\circ}\text{C}/\text{min}$. Raw data for absorbance at 260 nm (Figure 2b) show two transitions for

each sample: a transition between roughly 45 and 60 $^{\circ}\text{C}$ where individual tiles are presumed to form and a strongly hysteretic transition between roughly 25 and 45 $^{\circ}\text{C}$ where tubes are presumed to form. All transitions are several degrees lower for the SEJ-N1 tile than for the VE-N0 tile. (These formation and melting curves reveal a number of unusual characteristics, such as the hysteresis of the tile formation and the large size of the tube formation transition, that are the subject of a separate study. Hysteresis between tube formation and melting is expected as a consequence of a nucleation barrier, assembly of the smallest full-circumference tube.) Although the absolute temperature at which significant tube nucleation and growth occur will depend on the speed of cooling, these results suggest that during the annealing of COPY tubes, where both tiles are present in the same solution, VE-N0 single-tile tubes may nucleate and grow first, at a temperature too high for significant amounts of SEJ-N1 tiles to nucleate either as single-tile tubes or as heterogeneous COPY tubes.

Therefore, to observe nontrivial algorithmic growth on tubes—i.e., the copying of nontrivial bit patterns—we attempted to avoid disproportionation during sample preparation. Samples were annealed as before, but upon reaching 45 $^{\circ}\text{C}$, the temperature was dropped rapidly to 20 $^{\circ}\text{C}$ for 3 min to nucleate tubes, then raised again to 35 $^{\circ}\text{C}$, and cooled to 20 $^{\circ}\text{C}$ at 1 $^{\circ}\text{C}/\text{min}$. The intention was to favor legal tube growth after fast random nucleation of seed assemblies at the lower temperature; legal growth is most likely near the tube melting temperature where legal tile additions are thermodynamically favorable while illegal tile additions are thermodynamically unfavorable.^{17,31} This protocol results in a higher frequency of tubes with stripe patterns, as well as an increased fraction of assemblies with undefined character.

When AFM imaging of open tubes permits identification of individual tiles, we observe stripes of the hairpin-labeled SEJ-N1 tiles, indicating that the binary pattern of tiles was successfully copied from layer to layer during tube growth (Figure 2d). Stripes frequently “wrap around” from one side to the other, suggesting that they formed a contiguous helical stripe on the intact closed tube, that few tiles are lost during the opening process, and that few tiles were gained by assembly while on the mica. The stripe period can also give an indirect measurement of tube circumference (in tiles). However, we seldom see stripes continue for more than 10 layers without an error changing the stripe pattern—either losing or adding a stripe or several stripes. This is consistent with the 1–10% per-tile error rate observed in previous algorithmic self-assembly experiments with similar tiles.¹⁷ It is also consistent with tube joining processes that have been observed in single-tile DAE-E tubes;³⁸ here, COPY tubes that nucleate with different bit patterns may also join end-to-end if sufficiently many sticky ends match, resulting in apparent copying errors.

To test the full set of all four rule tiles, which is not expected to give clearly interpretable results when annealed to form tubes and assemblies, we first created a scaffold corresponding to the blue column in the abstract assembly diagram (Figure 1b). The procedure is a variation of that used in ref 17 and results in the desired repetitive single-stranded DNA product as well as some undesired repetitive double-stranded product. The total concentration of repeat unit (in either form) was estimated by UV260 absorbance after phenol/chloroform extraction and ethanol precipitation, and a stock was prepared at a nominal 300 nM. An estimate of the length of the scaffold strands can be obtained by annealing the scaffold strands with the component strands for the scaffold tile and tiles UE-C1 and REJ-C0, which should be able to form just a single layer of tiles on the scaffold. Each rule and scaffold tile component strand concentration was 200 nM, while the scaffold repeat unit concentration was nominally 25 nM. As imaged by AFM, the resulting linear structures typically range from 50 to 500 nm in length (Figure 2e, inset). (They not appear clearly perhaps due to insufficient adhesion to the mica surface.) This was deemed sufficient for attempting to observe binary counting crystallization to the extent that it will occur given the expected error rates for algorithmic self-assembly with this type of DNA tile.

For annealing of the scaffold strand, scaffold tiles, and all four rule tiles, we chose a schedule that cools from 90 to 50 °C at 1 °C/min, then proceeds from 50 to 30 °C at 0.1 °C/min, and finally decreases from 30 to 20 °C at 1 °C/min. On the basis of the thermal profiles of Figure 2b, we estimate that scaffold tiles and rule tiles should be completely formed by 45 °C and that nucleation of rule tiles on the scaffold assembly should begin to occur just below the crystal melting temperature (and thus prior to tube formation). The anneal proceeds slowly through this critical phase, during which we presume that algorithmic self-assembly occurs. Rule tile and scaffold tile component strands were, as before, at 200 nM, while the scaffold repeat

unit was at a nominal 25 nM; if 10% of the scaffold material was single-stranded, uniform growth exclusively on the scaffold would complete 80 layers before exhausting the VE-N0 rule tile. AFM imaging revealed a mixture of qualitatively different DNA assemblies (Figure 2e). Unfortunately, COPY tubes and all-0 tubes, apparently nucleated without the scaffold strand, were common, as were ill-defined assemblies that we have not been able to clearly interpret. Searching revealed crystals that had one straight edge, suggesting the presence of the scaffold strand, and a pattern of bits along that edge that suggested pattern formation according to the binary counting algorithm. Closer examination of these patterns revealed errors that disrupt the ideal pattern. Figure 3a shows five of the clearest crystals with mostly intact binary counting patterns. We interpret the images as tile assemblies as shown in Figure 3b; unclear imaging and lattice defects (missing tiles as well as inserted rows and columns) prevent unambiguous interpretation at some locations. According to the interpretation, there are six perfect countings from 1 to 7, including two from 0 to 8, and an overall error rate of 22 mistakes in 486 tiles (4.5%). Given that we examined roughly 75 scaffold-nucleated crystals (many at a resolution too poor to identify tiles) with a total length of no more than 33 μm , i.e., 2400 tiles, it is exceedingly unlikely that we would have observed any counting patterns similar to those in Figure 3a if the assembly had been random (i.e., an error rate of 50% during assembly). Had the error rate been as high as 15%, the probability of observing any domain of 126 tiles with fewer than four errors (such as the middle example) would have been less than 1% (see Supporting Information). Therefore, although we have no evidence that errors are independent, we are confident that algorithmic self-assembly is indeed occurring with an error rate no more than 15%, and we estimate that an error rate of 10% is more characteristic of the process.

Errors could be due to several types of (undesired) processes, in addition to simple growth errors. Misfolded tiles or tiles lacking some component strand could incorporate into the crystal and cause an error; this problem could be reduced by separately annealing and purifying each tile type. (Previous studies in our group¹⁷ estimated that typically between 80% and 95% of DAE-E tiles are well-formed.) Because the linear scaffold here relies on an insufficient attachment in order to initiate growth in each column layer, multiple initiation sites could conflict logically (as discussed above). Furthermore, initiation of the first layer of tiles at distance sites on a scaffold could result in a “hinged” crystal that, when it grows together, could contain many mismatch errors and lattice defects at the seam. Previously proposed tile sets for binary counters avoid these problems by containing an orthogonal structure (e.g., a row of 1’s) intended to quickly and controllably initiate growth into new layers. Finally, the most striking problem in the work reported here is that the overall yield is very low: by far the majority of the material is in the form of tubes, amorphous aggregates, and illegal growth from scaffold strands. Improved control of nucleation for algorithmic self-assembly, using more rigid nucleating structures, is needed

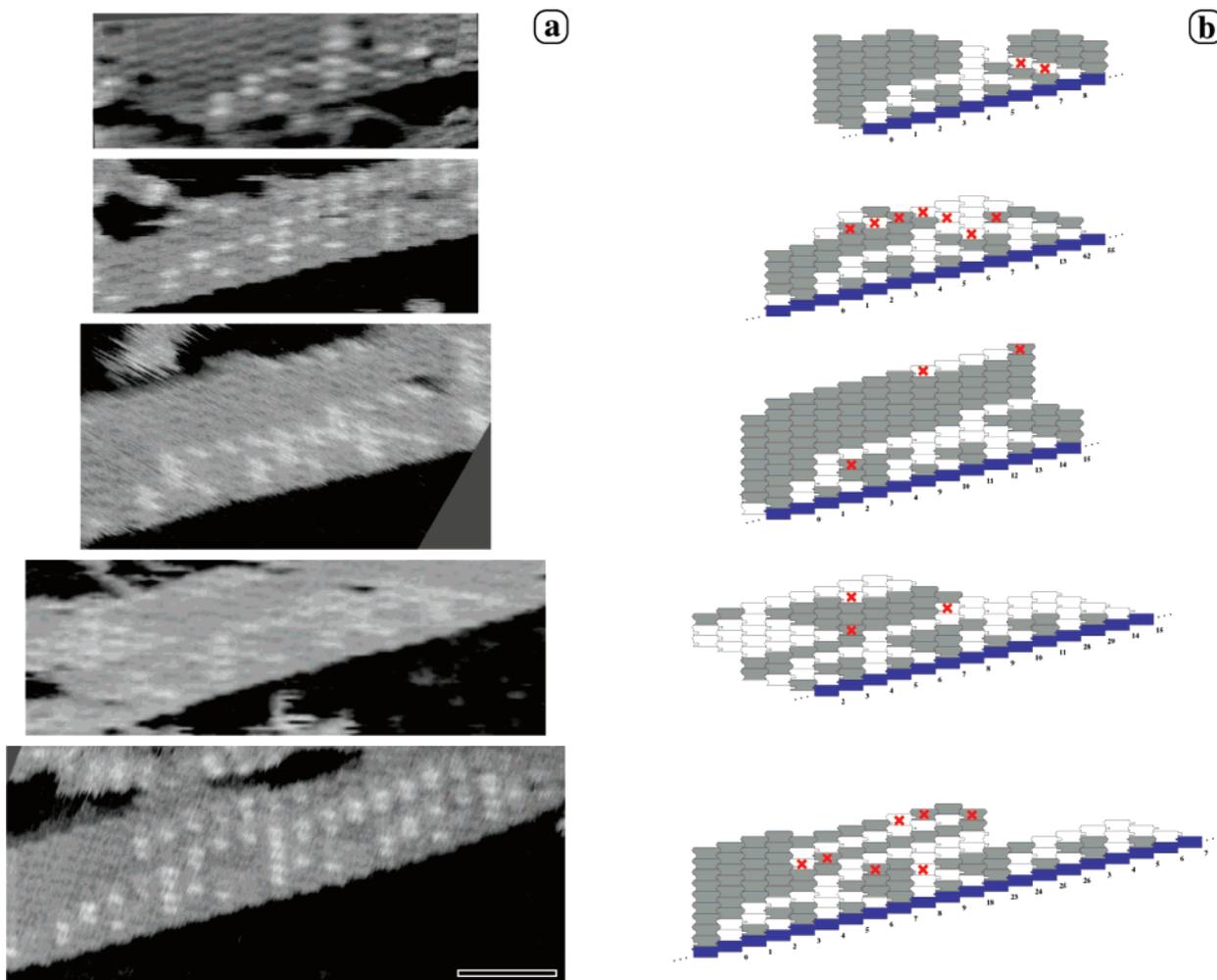


Figure 3. AFM images of good counting, and reasonable interpretations. (a) Five examples. Areas shown are selected from larger crystals that extend further to the left and/or right. The first one is an average of several scans of the same crystal. The last one has been “deghosted” to reduce AFM artifacts due to a double tip. Scale bar is 100 nm. (b) Interpretations of the images to the left. Red cross indicates tiles that mismatch their right or lower neighbor. Numbers give the binary integer represented by the nearby row; sometimes higher-order bits were ignored. Areas with missing tiles or lattice mismatches were not interpreted. Of 486 interpreted tiles, there were 22 errors, giving an overall 4.5% error rate.

to reduce these errors. Our thermodynamic study identified factors essential for obtaining this control. Unless the strengths of tile associations can be matched between tile types, so that the crystal melting temperature is independent of the tiles involved, annealing will first pass through a temperature at which only certain (usually erroneous) patterns can grow.

At the error rates demonstrated here, algorithmic self-assembly is not yet useful for technological applications such as creating scaffolds for arranging molecular electronic components^{20,21,23,39–41} or as templates for chemical postprocessing,^{42–44} because the largest error-free structures created contain fewer than 100 tiles, and other more reliable non-algorithmic methods are available at that scale.^{36,45–47} However, theory predicts that it should be possible to reduce nucleation errors⁴⁸ and growth errors^{33,49,50} to any required level, just by using similar DNA Wang tiles whose binding logic has been reprogrammed to incorporate simple error correcting codes. If experimental demonstration of these approaches, or improved tile designs or control of physical

conditions, can achieve error rates of 10^{-4} , then it will become possible to create assemblies of 10^4 tiles with specific algorithmically defined patterns. Reducing errors to this level remains an important challenge.

This work provides further evidence that algorithmic self-assembly provides a general mechanism for universal construction in the sense of von Neumann. The same self-assembly principles exploited here can easily be applied to implement other logical rules and are sufficient for simulating cellular automata, Turing machines, and other algorithmic logic for controlling crystal growth and patterning.^{14,25,51} Although there are a limited number of unique 5-mer sticky-ends that can be used to program DNA tiles with the DAE-E motif described here, it is already sufficient for implementing a number of universal Turing machines and cellular automata.^{52–54} Furthermore, different tile geometries and longer sticky ends can potentially increase the number of unique tiles that can be created. Thus the complexity of molecular objects synthesized by bottom-up fabrication appears to be limited only by the ability to reduce error rates

and by our cleverness in designing small developmental programs for algorithmic self-assembly of useful patterns.

Acknowledgment. For useful discussions we thank Rebecca Schulman and Deborah Fygenon. We thank the Caltech Molecular Materials Research Center for use of their AFM scanners. R.D.B. was partially supported by a Thomas E. Everhart Caltech Summer Undergraduate Research Fellowship (SURF). P.W.K.R. was supported by a Beckman Fellowship and Moore Center for the Physics of Information postdoctoral fellowship. E.W. acknowledges National Science Foundation awards 0093486 and 0432193.

Supporting Information Available: Experimental methods, diagrams for the DX molecules used, and sequences for all DNA strands. This information is available free of charge via the Internet at <http://pubs.acs.org>.

References

- (1) Dahiyat, B. I.; Mayo, S. L. *Science* **1997**, *278*, 82–87.
- (2) Ringler, P.; Schulz, G. E. *Science* **2003**, *302*, 106–109.
- (3) Seeman, N. C. *J. Theor. Biol.* **1982**, *99*, 237–247.
- (4) Seeman, N. C. *Biochemistry* **2003**, *42*, 7259–7269.
- (5) von Neumann, J. *Theory of Self-Reproducing Automata*; University of Illinois Press: Urbana, IL, 1966.
- (6) Whitesides, G. M.; Mathias, J. P.; Seto, C. T. *Science* **1991**, *254*, 1312–1319.
- (7) Lehn, J.-M. *Rep. Prog. Phys.* **2004**, *67*, 249–265.
- (8) Wang, H. An unsolvable problem on dominoes; Technical Report BL-30 (II-15), Harvard Computation Laboratory, 1962.
- (9) Grünbaum, B.; Shephard, G. C. *Tilings and Patterns*; W. H. Freeman and Co.: New York, 1987.
- (10) Winfree, E. On the Computational Power of DNA Annealing and Ligation. In *DNA Based Computers*; Lipton, R. J., Baum, E. B., Eds.; American Mathematical Society: Providence, RI, 1996; Vol. 27.
- (11) Reif, J. Local parallel biomolecular computing. In *DNA Based Computers III*; Rubin, H., Wood, D. H., Eds.; American Mathematical Society: Providence, RI, 1999; Vol. 48.
- (12) Adleman, L. M. Toward a mathematical theory of self-assembly; Computer Science Technical Report #00-722, University of Southern California, 2000.
- (13) Adleman, L.; Cheng, Q.; Goel, A.; Huang, M.-D.; Kempe, D.; de Espanes, P. M.; Rothmund, P. W. Combinatorial Optimization Problems in Self-Assembly. In *Proc. STOC* 2002.
- (14) Soloveichik, D.; Winfree, E. Complexity of self-assembled shapes. In *DNA Computing 10*; Ferretti, C., Mauri, G., Zandron, C., Eds.; Springer-Verlag: Berlin Heidelberg, 2005; Vol. LNCS 3384, Extended abstract; preprint of the full paper is [cs.CC/0412096](http://arxiv.org) on arXiv.org.
- (15) Adleman, L. M. *Science* **1994**, *266*, 1021–1024.
- (16) Mao, C.; LaBean, T. H.; Reif, J. H.; Seeman, N. C. *Nature* **2000**, *407*, 493–496.
- (17) Rothmund, P. W. K.; Papadakis, N.; Winfree, E. *PLoS Biol.* **2004**, *2*, e424.
- (18) Fu, T.-J.; Seeman, N. C. *Biochemistry* **1993**, *32*, 3211–3220.
- (19) Winfree, E.; Liu, F.; Wenzler, L. A.; Seeman, N. C. *Nature* **1998**, *394*, 539–544.
- (20) Collier, C. P.; Wong, E. W.; Belohradsky, M.; Raymo, F. M.; Stoddart, J. F.; Kuekes, P. J.; Williams, R. S.; Heath, J. R. *Science* **1999**, *285*, 391–394.
- (21) Huang, Y.; Duan, X.; Cui, Y.; Lauhon, L. J.; Kim, K.-H.; Lieber, C. M. *Science* **2001**, *294*, 1313–1317.
- (22) Cook, M.; Rothmund, P. W. K.; Winfree, E. Self-Assembled Circuit Patterns. In *DNA Computing 9*; Chen, J., Reif, J., Eds.; Springer-Verlag: Berlin, Heidelberg, 2004; Vol. LNCS 2943.
- (23) DeHon, A. *ACM J. Emerging Technol. Computing Syst.* **2005**, *1*, 109–162.
- (24) Beckman, R.; Johnston-Halperin, E.; Luo, Y.; Green, J. E.; Heath, J. R. *Science* **2005**, *310*, 465–468.
- (25) Rothmund, P. W. K.; Winfree, E. The Program-Size Complexity of Self-Assembled Squares. In *Symposium on Theory of Computing (STOC)*; ACM: New York, 2000.
- (26) Adleman, L. M.; Cheng, Q.; Goel, A.; Huang, M.-D. A. Running time and program size for self-assembled squares. In *ACM Symposium on Theory of Computing*; ACM: New York, 2001.
- (27) Aggarwal, G.; Goldwasser, M.; Kao, M.; Schweller, R. T. Complexities for generalized models of self-assembly. In *Symposium on Discrete Algorithms*; 2004.
- (28) Cairns-Smith, A. G. *J. Theor. Biol.* **1965**, *10*, 53–88.
- (29) Savage, J. E. *Models of Computation: Exploring the Power of Computing*; Addison-Wesley Longman, Inc.: Reading, MA, 1998; p 59.
- (30) Rothmund, P. W. K. Theory and Experiments in Algorithmic Self-Assembly. Thesis, University of Southern California, Department of Computer Science, 2001.
- (31) Winfree, E. Simulations of Computing by Self-Assembly; Technical Report CSTR: 1998.22, Caltech, 1998.
- (32) Winfree, E. *J. Biomol. Struct. Dyn.* **2000**, 263–270. Special issue S2.
- (33) Chen, H.-L.; Goel, A. Error Free Self-Assembly using Error Prone Tiles. In *DNA Computing 10*; Ferretti, C., Mauri, G., Zandron, C., Eds.; Springer-Verlag: Berlin, Heidelberg, 2005; Vol. LNCS 3384.
- (34) Rothmund, P. W. K.; Ekani-Nkodo, A.; Papadakis, N.; Kumar, A.; Fygenon, D. K.; Winfree, E. *J. Am. Chem. Soc.* **2004**, *126*, 16344–16352.
- (35) LaBean, T. H.; Winfree, E.; Reif, J. H. Experimental Progress in Computational Self-Assembly of DNA Tilings. In *DNA Based Computers V*; Winfree, E., Gifford, D. K., Eds.; American Mathematical Society: Providence, RI, 2000; Vol. 54.
- (36) Yan, H.; LaBean, T. H.; Feng, L.; Reif, J. H. *Proc. Natl. Acad. Sci. U.S.A.* **2003**, *100*, 8103–8108.
- (37) Stemmer, W. P. C.; Cramer, A.; Ha, K. D.; Brennan, T. M.; Heyneker, H. L. *Gene* **1995**, *164*, 49–53.
- (38) Ekani-Nkodo, A.; Kumar, A.; Fygenon, D. K. *Phys. Rev. Lett.* **2004**, *93*, 268–301.
- (39) Robinson, B. H.; Seeman, N. C. *Protein Eng.* **1987**, *1*, 295–300.
- (40) Bachtold, A.; Hadley, P.; Nakanishi, T.; Dekker, C. *Science* **2001**, *294*, 1317–1320.
- (41) Pease, A. R.; Jeppesen, J. O.; Stoddart, J. F.; Luo, Y.; Collier, C. P.; Heath, J. R. *Acc. Chem. Res.* **2001**, *34*, 433–444.
- (42) Alivisatos, A. P.; Johnsson, K. P.; Peng, X.; Wilson, T. E.; Loweth, C. J.; Bruchez, Jr., M. P.; Schultz, P. G. *Nature* **1996**, *382*, 609–611.
- (43) Le, J. D.; Pinto, Y.; Seeman, N. C.; Musier-Forsyth, K.; Taton, T. A.; Kiehl, R. A. *Nano Lett.* **2004**, *4*, 2343–2347.
- (44) Deng, Z. X.; Mao, C. D. *Angew. Chem., Int. Ed.* **2004**, *43*, 4068–4070.
- (45) Chworos, A.; Severcan, I.; Koyfman, A. Y.; Weinkam, P.; Oroudjev, E.; Hansma, H. G.; Jaeger, L. *Science* **2004**, *306*, 2068–2072.
- (46) Rothmund, P. W. K. *Nature*, in press.
- (47) Park, S.; Pistol, C.; Ahn, S.; Reif, J.; Lebeck, A.; Dwyer, C.; Labeau, T. *Angewandte Chemie*, in press.
- (48) Schulman, R.; Winfree, E. Programmable Control of Nucleation for Algorithmic Self-Assembly. In *DNA Computing 10*; Ferretti, C., Mauri, G., Zandron, C., Eds.; Springer-Verlag: Berlin, Heidelberg, 2005; Vol. LNCS 3384.
- (49) Winfree, E.; Bekbolatov, R. Proofreading Tile Sets: Error-Correction for Algorithmic Self-Assembly. In *DNA Computing 9*; Chen, J., Reif, J., Eds.; Springer-Verlag: Berlin, Heidelberg, 2004; Vol. LNCS 2943.
- (50) Reif, J. H.; Sahu, S.; Yin, P. Compact Error-Resilient Computational DNA Tiling Assemblies. In *DNA Computing 10*; Ferretti, C., Mauri, G., Zandron, C., Eds.; Springer-Verlag: Berlin Heidelberg, 2005; Vol. LNCS 3384.
- (51) Kao, M.-Y.; Ramachandran, V. DNA Self-Assembly For Constructing 3D Boxes. In *Algorithms and Computation: ISAAC 2001 Proceedings*; Springer-Verlag: Berlin, 2001; Vol. LNCS 2223.
- (52) Watanabe, S. *J. Assoc. Computing Machinery* **1961**, *8*, 476–483.
- (53) Lindgren, K.; Nordahl, M. *Complex Syst.* **1990**, *4*, 299–318.
- (54) Cook, M. *Complex Syst.* **2004**, *15*, 1–40.
- (55) *DNA Computing 10*; Ferretti, C., Mauri, G., Zandron, C., Eds.; Springer-Verlag: Berlin, Heidelberg, 2005; Vol. LNCS 3384.
- (56) *DNA Computing 9*; Chen, J., Reif, J., Eds.; Springer-Verlag: Berlin, Heidelberg, 2004; Vol. LNCS 2943.

NL052038L