JACS
JOURNAL OF THE AMERICAN CHEMICAL SOCIETY

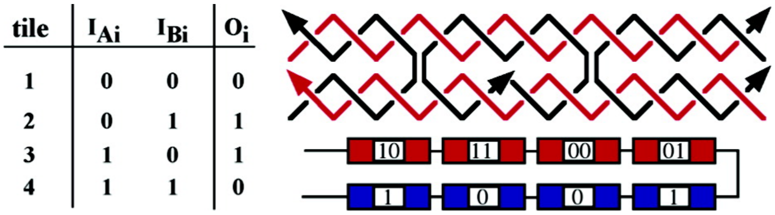
Communication

Parallel Molecular Computations of Pairwise Exclusive-Or (XOR) Using DNA “String Tile” Self-Assembly

Hao Yan, Liping Feng, Thomas H. LaBean, and John H. Reif

J. Am. Chem. Soc., **2003**, 125 (47), 14246-14247• DOI: 10.1021/ja036676m • Publication Date (Web): 30 October 2003

Downloaded from http://pubs.acs.org on March 30, 2009



tile	I_{Ai}	I_{Bi}	O_i
1	0	0	0
2	0	1	1
3	1	0	1
4	1	1	0

ACS Publications
High quality. High impact.

Journal of the American Chemical Society is published by the American Chemical Society. 1155 Sixteenth Street N.W., Washington, DC 20036

J|A|C|S
COMMUNICATIONS

Published on Web 10/30/2003

Parallel Molecular Computations of Pairwise Exclusive-Or (XOR) Using DNA "String Tile" Self-Assembly

Hao Yan,* Liping Feng, Thomas H. LaBean, and John H. Reif*

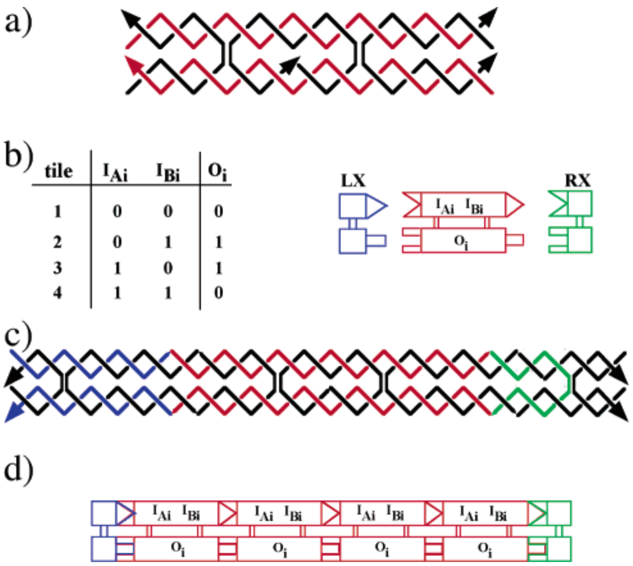
Department of Computer Science, Duke University, Durham, North Carolina 27708

Received June 13, 2003; E-mail: reif@cs.duke.edu; hy1@cs.duke.edu

DNA computing[1] potentially provides a degree of parallelism far beyond that of conventional silicon-based computers. A number of researchers[2] have experimentally demonstrated DNA computing in solving instances of the satisfiability problems. Self-assembly of DNA nanostructures is theoretically an efficient method of executing parallel computation where information is encoded in DNA tiles and a large number of tiles can be self-assembled via sticky-end associations. Winfree et al.[3] have shown that representations of formal languages can be generated by self-assembly of branched DNA nanostructures and that two-dimensional DNA self-assembly is Turing-universal (i.e., capable of universal computation). Mao et al.[4] experimentally implemented the first algorithmic DNA self-assembly which performed a logical computation (cumulative exclusive-or (XOR)); however, that study only executed two computations on fixed input. To further explore the power of computing using DNA self-assembly, experimental demonstrations of parallel computations are required. Here, we describe the first parallel molecular computation using DNA tiling self-assembly in which a large number of distinct input are simultaneously processed.

The experiments described here are based on the "string tile" model proposed by Winfree et al.,[5] where they investigated computation by linear assemblies of complex DNA tiles. The concept of "string tile" assemblies derives from Eng's observation[6] that, by allowing neighboring tiles in an assembly to associate by sticky ends on each side, one could increase the computational complexity of languages generated by linear self-assemblies. Surprisingly sophisticated calculations can be performed with single-layer linear assemblies when contiguous strings of DNA trace through individual tiles and the entire assembly multiple times. In essence, a "string tile" is the collapse of a multilayer assembly into a simpler superstructure by allowing individual tiles to carry multiple segments of the reporter strands, thereby allowing an entire row of a truth table to be encoded within each individual tile. "String tile" arithmetic implementations have a number of advantageous properties. (i) Input and output strings assemble simultaneously. (ii) Each row in the truth table for the function being calculated is represented as a single tile type, where all input and output bits are encoded on that tile. Each pairwise operation is directly encoded in the structure of a tile.

We have experimentally implemented "string tile" arithmetic using linear self-assembly of DNA double crossover tiles[7] (DX) to perform pairwise XOR calculations. The molecular structure of the DX tiles used here is illustrated in Figure 1a. It contains five strands that self-assemble through Watson−Crick base pairing to produce two double helices which are connected to each other at two points where their strands cross over between them. There are two continuous strands (red) going in opposite directions through the DX tile, and they are used to encode input and output information. Figure 1b gives the truth table for XOR and a representation of the computational tile types. The two helical domains are drawn as rectangles, flanked by sticky ends shown as geometrical shapes.



tile	I_{Ai}	I_{Bi}	O_i
1	0	0	0
2	0	1	1
3	1	0	1
4	1	1	0

Figure 1. Parallel computing of pairwise XOR by self-assembly of DNA tiles.

The two input values of each tile are in the upper rectangle, and the output value is in the lower rectangle. The corner tiles are labeled as LX and RX. One tile type is required for each of the four rows in the truth table, plus one tile type is needed for each of the two ends of the assembly. The two input bits are encoded with sequence words between the two crossover points on the upper continuous strand. The output bit is encoded on the lower continuous strand. In parallel computation of pairwise XOR, multiple DX tiles will self-assemble into multitile linear superstructures via sticky-end associations. For the purpose of generating a reporter strand which connects a string of input with its output, a right corner tile (RX) is introduced. This corner tile is a four-arm junction,[8] which links the input strand in the top helical domain to the output strand in the bottom helical domain. Another four-arm junction corner tile (LX) is used to introduce primer-binding sites for polymerase chain reaction (PCR) amplification of the reporter strands. Figure 1c shows an example of the reporter strand structure with one computational tile and two corner tiles. Figure 1d shows an example of a tile assembly capable of calculating 4-bit XOR. In this design, I_{Ai} and I_{Bi} indicate sequences encoding binary values for the two input bits of the current operation. O_i encodes the value of the output bit. The linear assembly shown in Figure 1d is composed of six tiles, starting on the left with an LX tile, followed by four computational tiles, and ending with an RX tile. The ordering of I/O bit values (from 5′ to 3′) on the reporter strand is as follows: I_AI_B (first input bit-pair through *n*th input bit-pair, interleaved), O (*n*th output bit through first output bit). This orientation allows the input to be read with the lowest bit first (on

10.1021/ja036676m CCC: $25.00 © 2003 American Chemical Society

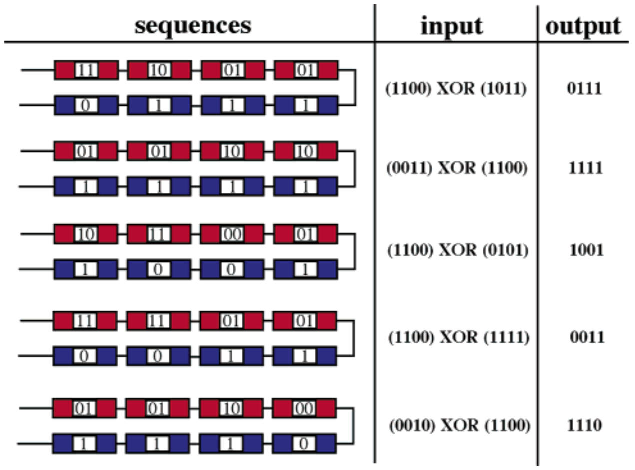


Figure 2. Simplified representation of the sequencing readout for parallel computation of pairwise XOR of "string tile" self-assembly.

the left), while the output bits are listed on the reporter strand in the opposite direction from the input.

Experimental execution of parallel XOR computations proceeded as follows: first, the set of purified oligonucleotides for each individual tile type were slowly annealed in separate tubes in T_4 DNA ligase buffer from 95 to 65 °C to ensure the formation of valid tiles before parallel self-assembly of multitile superstructures. Stalling the annealing at 65 °C (the approximate melting temperature of individual tiles) reduces the formation of multitile assemblies with repeated copies of one computational tile type while keeping the tiles in approximately their correct structures. Separately annealed computational and corner tiles were then mixed together at 65 °C and further annealed to 16 °C. The reporter strand segments were then ligated to one another to produce reporter strands that contain the input and output of the calculation. The estimated yield of the doubly ligated DX molecules is ∼7%. Readout of the calculations was accomplished by first selectively purifying ligated reporter strands corresponding to 4-bit computations (602 nucleotides) by extracting the band from a denaturing polyacrylamide gel of the ligation products. The purified reporter strands (containing 2^4 possible calculations) were then amplified by PCR. Purified PCR products of the proper length were then ligated with cloning vector PSTBlue-1 (Novagen, Madison, WI) and cloned (>200 clones were observed). Finally, dideoxy sequencing was performed on miniprep DNA from five randomly selected clones.

The sequencing results obtained from the five randomly selected clones match the designed sequence words and encode valid computations. Figure 2 summarizes the sequencing results obtained from the pairwise XOR calculation. For simplicity, the input and output read out from the sequencing results are illustrated as rectangles labeled with their corresponding bit values. The input values are recorded with the first bit on the left, while the output bits are listed in the opposite orientation. The corresponding pairwise XOR calculations are shown on the right. No strand mismatches (which would cause errors in the parallel calculations) were observed in any of the 20 computational tiles examined in the sequencing results. The five computational assemblies displayed completely different input strings and an adequate mixture of all four tile types, suggesting that the experiment successfully achieved the desired parallelism. We have demonstrated the prototype system by sequencing some example 4-bit calculations. The ligation experiment indicates that we were able to obtain reporter strands up to about 15 computational tiles (∼1890 bp); thus the computation step resulted in a considerable degree of parallelism. However, problems with the PCR amplification prevented readout of these longer computations. To realize the potential massive parallelism inherent in the present system, we must improve the ligation and readout efficiency for longer bit strings. Atomic force microscope (AFM) imaging of the unligated self-assembled tiles reveals that up to 25 computational tiles can be self-assembled together. Development of visual readout methods, where input and output can be distinguished using, for example, AFM by incorporating topographic markers in the set of computational tiles, will significantly enhance the readout efficiency of the massively parallel molecular computation.

We have demonstrated, for the first time, parallel computation using DNA tile self-assembly. In this system, the computational complexes randomly assemble and generate a molecular look-up table in which each reporter strand encodes the input and output of a valid calculation on a random input string. In principle, one could achieve the same result by simply ligating linear DNA duplex with sticky ends, terminated by a hairpin. The reason to choose DX tile over DNA duplex is that ligation of duplex terminated with a hairpin will result in a reporter strand that folds back to a single, long double helix, thus causing problems for PCR amplification. Also, ligation of linear DNA duplex assemblies commonly results in unwanted circular structures. Demonstration of parallel molecular computation in "string tile" assemblies using DX tiles, as well as the accuracy of the demonstrated computations, show the plausibility of more interesting "string tile" computations in the future. However, more efficient readout methods (such as visual readout) will need to be developed. The reporter strands generated in "string tile" self-assembly may be useful as input for further computations as they represent a unique combinatorial library of sequences with a complex structural theme.

Acknowledgment. This work has been supported by grants from NSF to H.Y., T.H.L., and J.H.R. (EIA-00-86015, EIA-0218376, EIA-0218359) and DARPA/AFSOR to J.H.R. (F30602-01-2-0561).

Supporting Information Available: Reporter strands ligation, AFM image of unligated assembly, PCR amplification of reporter strands, cloning of amplified reporter strands, and details of sequencing results (PDF). This material is available free of charge via the Internet at http://pubs.acs.org.

References

(1) (a) Adleman, L. M. *Science* **1994**, *266*, 1021−1024. (b) Lipton, R. J. *Science* **1995**, 268, 542−545. (c) Reif, J. H. *Science* **2002**, *296*, 478−479.

(2) (a) Liu, Q.; et al. *Nature* **2000**, *403*, 175−179. (b) Faulhammer, D.; Cukras, A. R.; Lipton, R. J.; Landweber, L. F. *Proc. Natl. Acad. Sci. U.S.A.* **2000**, *97*, 1385−1389. (c) Braich, R. S.; et al. *Science* **2002**, *296*, 499−502.

(3) Winfree, E. In *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*; Lipton, R., Baum, E. B., Eds.; Proceedings of the 1st DIMACS Workshop on DNA Based Computers: Princeton, 1995; Vol. 27, pp 199−221.

(4) Mao, C.; LaBean, T. H.; Reif, J. H.; Seeman, N. C. *Nature* **2000**, *407*, 493−496.

(5) Winfree, E.; Eng, T.; Rozenberg, G. In *DNA Computing*; Condon, A., Rozenberg, G., Eds.; Taylor and Francis: New York, 2001; pp 63−88.

(6) Eng, T. In *DNA Based Computers III: DIMACS Workshop, June 23−25, 1997*; Rubin, H., Wood, D. H., Eds.; American Mathematical Society, 1999.

(7) (a) Fu, T.-J.; Seeman, N. C. *Biochemistry* **1993**, *32*, 3211−3220. (b) Winfree, E.; Liu, F. L.; Wenzler, A.; Seeman, N. C. *Nature* **1998**, *394*, 539−544.

(8) Kallenbach, N. R.; Ma, R. I.; Seeman, N. C. *Nature* **1983**, *305*, 829−831.

JA036676M