JACS
JOURNAL OF THE AMERICAN CHEMICAL SOCIETY

Subscriber access provided by American Chemical Society

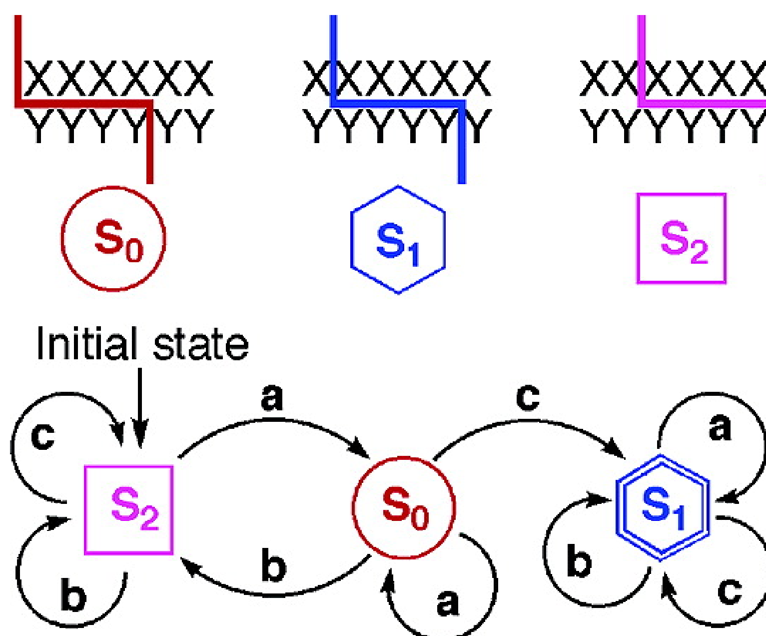
Article

Parallel Biomolecular Computation on Surfaces with Advanced Finite Automata

Michal Soreni, Sivan Yogev, Elizaveta Kossoy, Yuval Shoham, and Ehud Keinan

J. Am. Chem. Soc., **2005**, 127 (11), 3935-3943• DOI: 10.1021/ja047168v • Publication Date (Web): 23 February 2005

Downloaded from http://pubs.acs.org on March 24, 2009



More About This Article

Additional resources and features associated with this article are available within the HTML version:

- Supporting Information
- Links to the 2 articles that cite this article, as of the time of this article download
- Access to high resolution figures
- Links to articles and content related to this article
- Copyright permission to reproduce figures and/or text from this article

View the Full Text HTML

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
ARTICLES
Published on Web 02/23/2005

Parallel Biomolecular Computation on Surfaces with Advanced Finite Automata

Michal Soreni,[†] Sivan Yogev,[†] Elizaveta Kossoy,[†] Yuval Shoham,*[,‡] and Ehud Keinan*[,†,§]

Contribution from the Department of Chemistry, and Department of Biotechnology and Food Engineering, Institute of Catalysis Science and Technology, Technion − Israel Institute of Technology, Technion City, Haifa 32000, Israel, and Department of Molecular Biology and The Skaggs Institute for Chemical Biology, The Scripps Research Institute, 10550 North Torrey Pines Road, La Jolla, California 92037

Received May 14, 2004; E-mail: keinan@techunix.technion.ac.il

Abstract: A biomolecular, programmable 3-symbol-3-state finite automaton is reported. This automaton computes autonomously with all of its components, including hardware, software, input, and output being biomolecules mixed together in solution. The hardware consisted of two enzymes: an endonuclease, *Bbv*I, and T4 DNA ligase. The software (transition rules represented by transition molecules) and the input were double-stranded (ds) DNA oligomers. Computation was carried out by autonomous processing of the input molecules via repetitive cycles of restriction, hybridization, and ligation reactions to produce a final-state output in the form of a dsDNA molecule. The 3-symbol-3-state deterministic automaton is an extension of the 2-symbol-2-state automaton previously reported, and theoretically it can be further expanded to a 37-symbol-3-state automaton. The applicability of this design was further amplified by employing surface-anchored input molecules, using the surface plasmon resonance technology to monitor the computation steps in real time. Computation was performed by alternating the feed solutions between endonuclease and a solution containing the ligase, ATP, and appropriate transition molecules. The output detection involved final ligation with one of three soluble detection molecules. Parallel computation and stepwise detection were carried out automatically with a Biacore chip that was loaded with four different inputs.

Introduction

In fully autonomous molecular computing devices, all components, including input, output, software, and hardware, are represented by specific molecules that interact with each other through a cascade of programmable chemical events, progressing from the input molecule to the molecular output signal.[1,2] DNA molecules and DNA enzymes have been employed as convenient, readily available components of such computing devices because they offer highly predictable recognition patterns, reactivity, and information-encoding features.[3] Furthermore, DNA-based computers can become part of a biological system, generating outputs in the form of biomolecular structures and functions.[4]

Our previously reported 2-symbol-2-state finite automata were computed autonomously with all of their components being soluble biomolecules mixed in solution.[1] The hardware consisted of two enzymes: a restriction nuclease and a ligase, with the software (transition rules represented by transition molecules) and the input being double-stranded (ds) DNA oligomers (Figure 1).[5] Programming was achieved by the choice of transition molecules that were mixed in solution. Computation was carried out by processing the input molecules via repetitive cycles of restriction, hybridization, and ligation reactions to produce a

† Department of Chemistry, Technion.
‡ Department of Biotechnology and Food Engineering, Technion.
§ The Scripps Research Institute.

(1) Benenson, Y.; Paz-Elizur, T.; Adar, R.; Keinan, E.; Livneh, Z.; Shapiro, E. *Nature* **2001**, *414*, 430−434.
(2) (a) Mao, C.; LaBean, T. H.; Relf, J. H.; Seeman, N. C. *Nature* **2000**, *407*, 493−496. (b) Rose, J. A.; Deaton, R. J.; Hagiya, M.; Suyama, A. *Phys. Rev. E* **2002**, *65*, 021910. (c) Sakamoto, K.; Gouzu, H.; Komiya, K.; Kiga, D.; Yokoyama, S.; Yokomori, T.; Hagiya, M. *Science* **2000**, *288*, 1223−1226. (d) Komiya, K.; Sakamoto, K.; Gouzu, H.; Yokoyama, S.; Arita, M.; Nishikawa, A.; Hagiya, M. In *DNA Computing: 6th International Workshop on DNA-Based Computers, DNA 2000 Leiden, The Netherlands, June 13−17, 2000*; Condon, A., Rozenberg, G., Eds.; Springer-Verlag: Berlin, 2001; pp 19−26. (e) Benenson, Y.; Adar, R.; Paz-Elizur, T.; Livneh, Z.; Shapiro, E. *Proc. Natl. Acad. Sci. U.S.A.* **2003**, *100*, 2191−2196. (f) Benenson, Y.; Gil, B.; Ben-Dor, U.; Adar, R.; Shapiro, E. *Nature* **2004**, *429*, 423−429.
(3) (a) Adleman, L. M. *Science* **1994**, *266*, 1021−1024. (b) Braich, R. S.; Chelyapov, N.; Johnson, C.; Rothemund, P. W.; Adleman, L. *Science* **2002**, *296*, 499−502. (c) Faulhammer, D.; Cukras, A. R.; Lipton, R. J.; Landweber, L. F. *Proc. Natl. Acad. Sci. U.S.A.* **2000**, *97*, 1385−1389. (d) Ouyang, Q.; Kaplan, P. D.; Liu, S.; Libchaber, A. *Science* **1997**, *278*, 446−449. (e) Yan, H.; Feng, L.; LaBean, T. H.; Reif, J. H. *J. Am. Chem. Soc.* **2003**, *125*, 14246−14247. (f) Yan, H.; LaBean, T. H.; Feng, L.; Reif, J. H. *Proc. Natl. Acad. Sci. U.S.A.* **2003**, *100*, 8103−8108.
(4) (a) Kossoy, E.; Soreni, M.; Shoham, Y.; Keinan, E. A Nanoscale programmable computing machine with molecular input, molecular software and bio-functional output. 16th International Conference on Physical Organic Chemistry, San Diego, CA, August, 2002. (b) Kossoy, E.; Soreni, M.; Shoham, Y.; Keinan, E. Toward Computing Machines Made of Biomolecules. Proceedings of the 6th World Multi-Conference on Systemic, Cybernetics and Informatics, Orlando, FL, SCI 2002/ISAS, 2002; Vol. XVII. (c) Kossoy, E.; Soreni, M.; Shoham, Y.; Keinan, E. A Nanoscale Programmable Computing Machine with Molecular Input, Molecular Software and Bio-Functional Output. The 1st Symposium of Bio-Inspired Engineering, Haifa, Israel, December, 2003.
(5) Rothemund, P. W. K. In *DNA based computers: Proceedings of the DIMACS workshop, Princeton University, April 4, 1995*; Lipton, R. J., Baum, E. B., Eds.; American Mathematical Society: Providence, RI, 1996; pp 75−119.

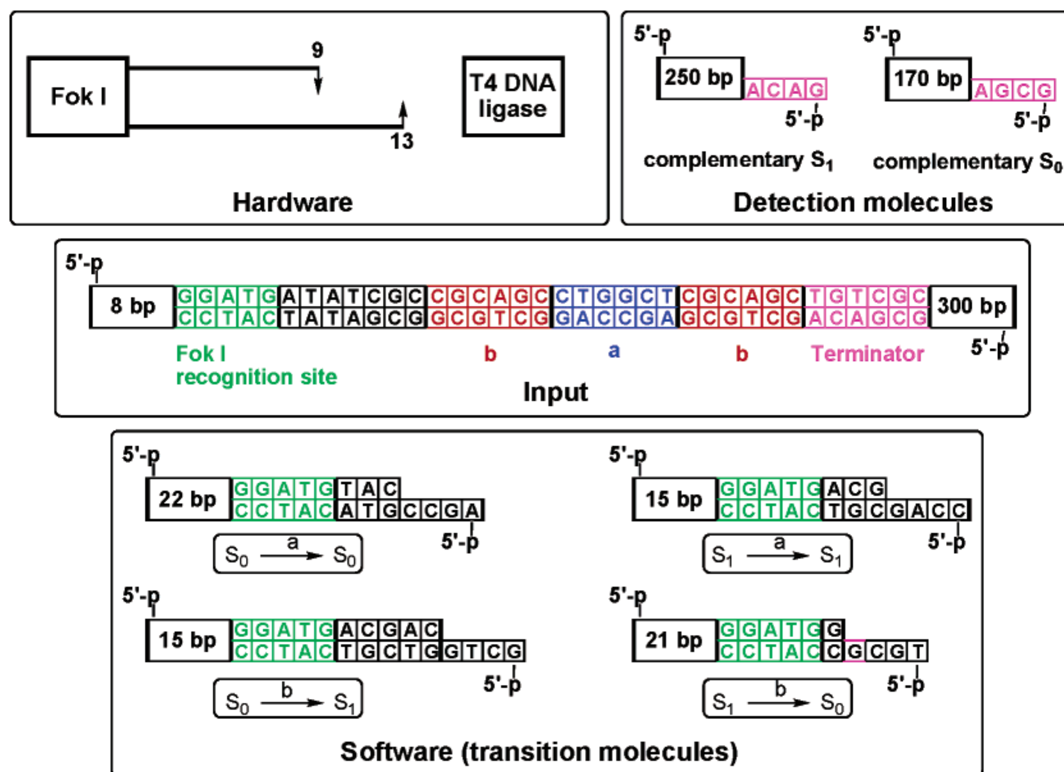


Figure 1. Design of a finite automaton with two symbols, **a** and **b**, and two states, S_0 and S_1. The 10-component automaton is comprised of an input molecule, two enzymes, ATP, four transition molecules, and two detection molecules. The four transition molecules shown here, which represent one specific program, are chosen from a library of eight transition molecules.

final-state output in the form of a dsDNA molecule that was identified by its length. Each symbol in the input molecule (**a** or **b**) was represented by a 6-base-pair (bp) sequence. Each state (S_1 or S_0) was represented by the depth cut into the symbol domain (Figure 1). While restriction at the beginning of the domain represented S_1, cutting 2 bp deeper into this domain represented S_0.

Although 2-symbol-2-state automata represent a proof of concept for autonomous computing, it has a limited computational power. To enhance this property, we considered two possible strategies: the first increases the level of complexity by addition of states and/or symbols, while the other focuses on parallel computation techniques.[6] Here, we present our progress in both directions, demonstrating an ability to carry out more complex computations by the design and realization of 3-symbol-3-state automata. Moreover, we show that immobilization of the input molecules allows for parallel computation, where the input location on a chip represents specific tagging. In addition, we show that the use of surface plasmon resonance (SPR) technology allows for real-time detection of the output signal as well as for real-time monitoring of all computation intermediates.

Results and Discussion

We reasoned that a 4-cutter endonuclease, such as *Fok*I or *Bbv*I, which cuts either 9 and 13 bases or 8 and 12 bases away from their recognition sites, respectively, could restrict a 6-bp symbol into three distinguishable modes: at the beginning of the symbol domain, 1 bp deeper, or 2 bp deeper into that domain. These three restriction modes could represent three different internal states, S_0, S_1, or S_2, respectively (Figure 2). Accordingly, a three-state automaton with two symbols would have a total library of 12 transition rules with a broader spectrum of possible programs, as compared to the previously reported 2-symbol-2-state case.[1] If the number of symbols could also be increased, the library of transition rules would increase dramatically because programming amounts to the selection of a subset from the full library of transition rules and the determination which internal states are accepting. For example, a 3-symbol-3-state device has a library of 27 possible transition rules. Because there are seven possible selections of the accepting states (S_0, S_1, S_2, any combination of two and a combination of all three), the overall number of syntactically distinct programs is 137 781. This figure is remarkably larger than the corresponding number of 48 possible programs offered by our previously reported 2-symbol-2-state device.[1]

The design of three symbols and a terminator requires that all 12 possible sticky ends, which are produced in the various modes of restriction, will be mutually exclusive. An example of such an array of symbols, **a**, **b**, **c**, and a terminator, **t**, as well as their complete set of possible restriction products, are shown in Figure 3. In comparison, our previously reported 2-symbol-2-state devices consisted of only six mutually exclusive sticky ends.[1]

In fact, the number of possible different symbols offered by DNA is much larger than 3 because there are $4^4 = 256$ options to create a 4-bp sticky end. Palindrome sticky ends cannot be

(6) (a) LaBean, T. H.; Winfree, E.; Reif, J. H. In *DNA based computers V: Proceedings of the DIMACS workshop, Massachusetts Institute of Technology, June 14−15, 1999*; Winfree, E., Gifford, D. K., Eds.; American Mathematical Society: Providence, RI, 2000; pp 123−140. (b) Roweis, S.; Winfree, E.; Burgoyne, R.; Chelyapov, N. V.; Goodman, M. F.; Rothemund, P. W. K.; Adleman, L. M. *J. Comput. Biol.* **1998**, *5*, 615−629.

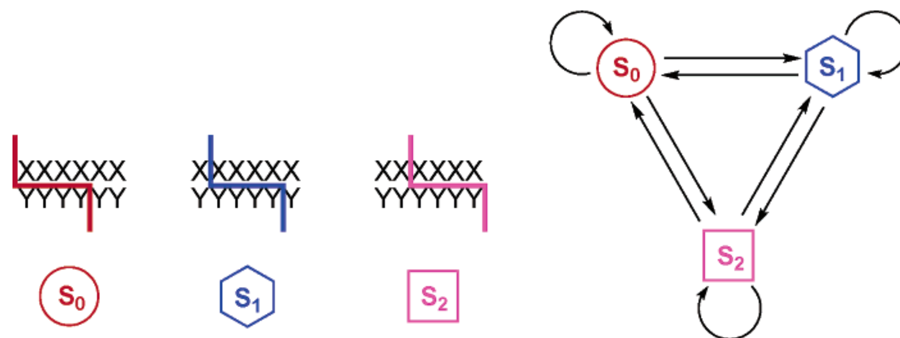


Figure 2. Three restriction modes of a 6-bp domain by a 4-cutter enzyme (left) represent three internal states and a total of 27 possible transition rules (shown graphically on the right).

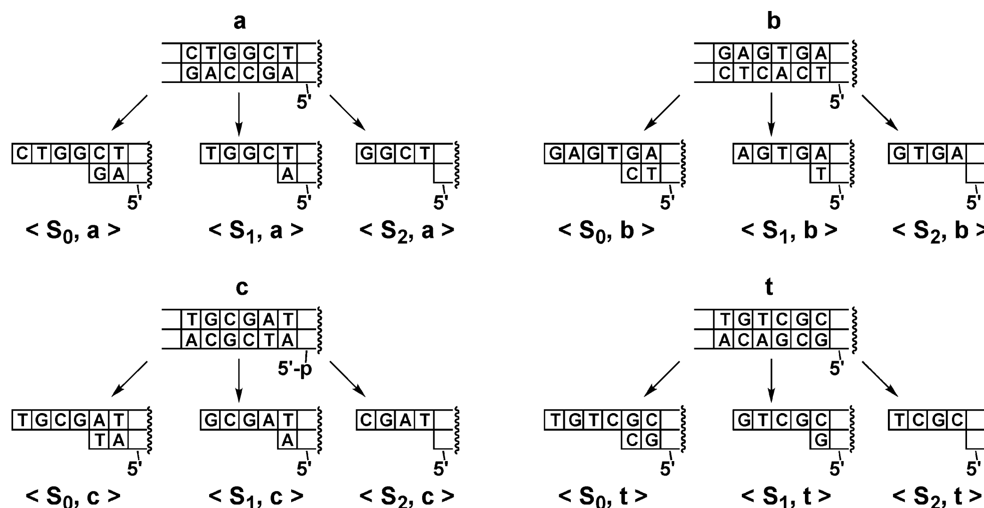


Figure 3. Three symbols, **a**, **b**, **c**, and terminator **t** with their restriction products produced by a 4-cutter.

used because they would stick to one another to produce centrosymmetric dimers. Therefore, as the number of all possible 4-bp palindromes is $4^2 = 16$, the total number of possible sticky ends that can be used in our system is reduced to 240. Each symbol that consists of 6 bp can be restricted in one of three different ways to produce one of three different sticky ends. Because each of these three sticky ends requires the availability of a transition molecule that has a complementary sticky end, the maximal number of possible orthogonal symbols is 240/6 = 40. This number suggests that automata with up to 39 symbols and 1 terminator can be realized on the basis of this technology. Table 1 shows the maximal set of distinct 6-bp sequences we found so far, which includes 38 sequences that can be used as 37 symbols and 1 terminator.[7]

An example of a finite automaton that has three symbols and three states is outlined in Figure 4, both graphically and in the form of a table consisting of nine transition rules. This automaton accepts the internal state S_1, which means that it finds input strings that contain the sequence **ac**, and therefore it accepts the formal language $\Sigma^*\mathbf{ac}\Sigma^*$, where $\Sigma = \{a,b,c\}$.

Immobilization of one or more components of this multicomponent system on a chip offers attractive opportunities.[5,8,9] For example, all reactions with an immobilized DNA molecule

Table 1. A List of 38 Sequences of Length 6 bp (Strand 1), with Their Complementary Sequences (Strand 2)[a]

No.	Strand 1	Strand 2	No.	Strand 1	Strand 2
1	AGTCAG	CTGACT	20	ACATTG	CAATGT
2	GAATCC	GGATTC	21	TCCCTC	GAGGGA
3	TTTGGG	CCCAAA	22	ACGACG	CGTCGT
4	GATAGC	GCTATC	23	GCAACG	CGTTGC
5	CGTAAG	CTTACG	24	AGTGAT	ATCACT
6	AGCGGG	CCCGCT	25	TCGGAG	CTCCGA
7	AAAGGC	GCCTTT	26	GGGGTA	TACCCC
8	GACCAC	GTGGTC	27	TCAAGC	GCTTGA
9	AGTTCG	CGAACT	28	CCAGCA	TGCTGG
10	CTGCCG	CGGCAG	29	AGACAG	CTGTCT
11	TAACAC	GTGTTA	30	TAGAAA	TTTCTA
12	GAGCCC	GGGCTC	31	CAGATG	CATCTG
13	CAGTAG	CTACTG	32	TTGTGC	GCACAA
14	TGTATG	CATACA	33	ACTCAT	ATGAGT
15	CTCGCA	TGCGAG	34	CAGGAC	GTCCTG
16	AAGAGA	TCTCTT	35	ATGGCG	CGCCAT
17	ACTTCC	GGAAGT	36	AACCTA	TAGGTT
18	ACTAAC	GTTAGT	37	ATAAAC	GTTTAT
19	CACGGT	ACCGTG	38	AAAATA	TATTTT

[a] These sequences are legal symbols in the described system, because no 4-bp sequence appears twice in the table, meaning that all possible sticky ends created during the calculation are unique to one state of one symbol.

may be driven to completion using the excess of soluble reactants. Further, the resultant immobilized DNA product may be easily purified by washing the chip with an appropriate solvent. More importantly, the use of immobilized inputs allows for parallel computation with many input molecules, every one of which is geographically labeled to allow specific correlation between input and output. These advantages maximize the reaction rates, efficiency, and fidelity of the computation process and minimize the error frequency. Corn and co-workers have

(7) This set was created for algorithmic reasons, and therefore it does not include the symbols used in our current design, which is shown in Figure 3. It is still an open question whether the maximal number of 6-bp sequences that produce distinct 4-bp sticky ends in both strands is 40.

(8) Gillmor, S. D.; Rugheimer, P. P.; Lagally, M. G. *Surf. Sci.* **2002**, *500*, 699−721.
(9) Sakakibara, Y.; Suyama, A. *Genome Inf. Ser.* **2000**, *11*, 33−42.

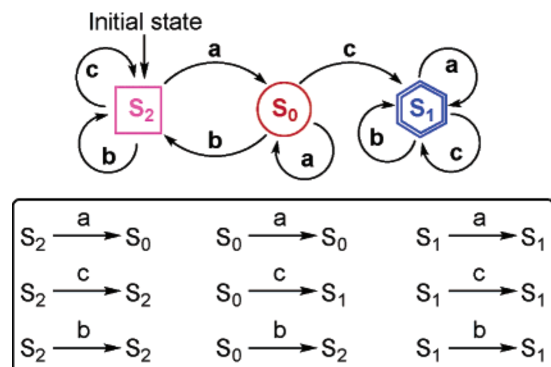


Figure 4. A 3-symbol-3-state finite automaton shown both graphically (top) and in the form of a table of nine transition rules (bottom).

already demonstrated many of these advantages by utilizing surface-based DNA for computation with a "word" design to store information in a string of bases.[10] Their operations, including hybridization of DNA words bound to solid surface, enzymatic digestion of ssDNA strands, and ligation, were used to solve a 3-SAT problem.[11]

In our design, each input molecule (Figure 5) contained a recognition site of the endonuclease *Bbv*I, which is known to cut the dsDNA at positions 8 and 12 away from its recognition site. Each input contained several symbols, 6-bp each, and a 6-bp terminator domain. We prepared our dsDNA input molecules with their 5′ position on the terminator end being biotinylated. This design allowed for that end to be attached to a streptavidin-coated Biacore chip.

The software in our system is represented by a chosen set of nine transition molecules (Figure 6), each containing a 5-bp long recognition site of *Bbv*I, a 4-bp sticky end, and a 0−4-bp spacer between these two domains.

Output detection was carried out by the use of three soluble detection molecules (Figure 7), each possessing a complementary sticky end to match one of the three possible sticky ends that are produced by restriction of the terminator domain. Thus, for example, detection molecule D−S_0 could undergo hybridization and ligation with the sticky end that represents the S_0 output, thereby creating a positive SPR response using the Biacore instrumentation.

Although in our previously reported homogeneous system all components were placed in a single mixture,[1] we found it advantageous to separate them into two mixtures. The first and most significant advantage was the ability to monitor the individual computation steps while carrying out parallel computation with multiple input molecules all bound to a single chip. The second advantage came up from the fact that, although none of the transition molecules was sufficiently long to become a substrate of *Bbv*I, they were all reversible inhibitors of this enzyme. Placing the enzyme and transition molecules in separate mixtures prevented this inhibition. The third advantage was related to the fact that *Bbv*I and T4 DNA ligase require different conditions for optimal efficiency. In our previously reported work, where both restriction enzyme and ligase were employed in a single mixture, we have made compromises in choosing the conditions to partially satisfy both enzymes. Here, the use of immobilized input molecules allowed for convenient separation of the two enzymes, permitting each to operate under its optimal conditions. We found that washing the system with a solution of SDS before switching from one mixture to another removed nonspecific interactions and provided smooth baselines

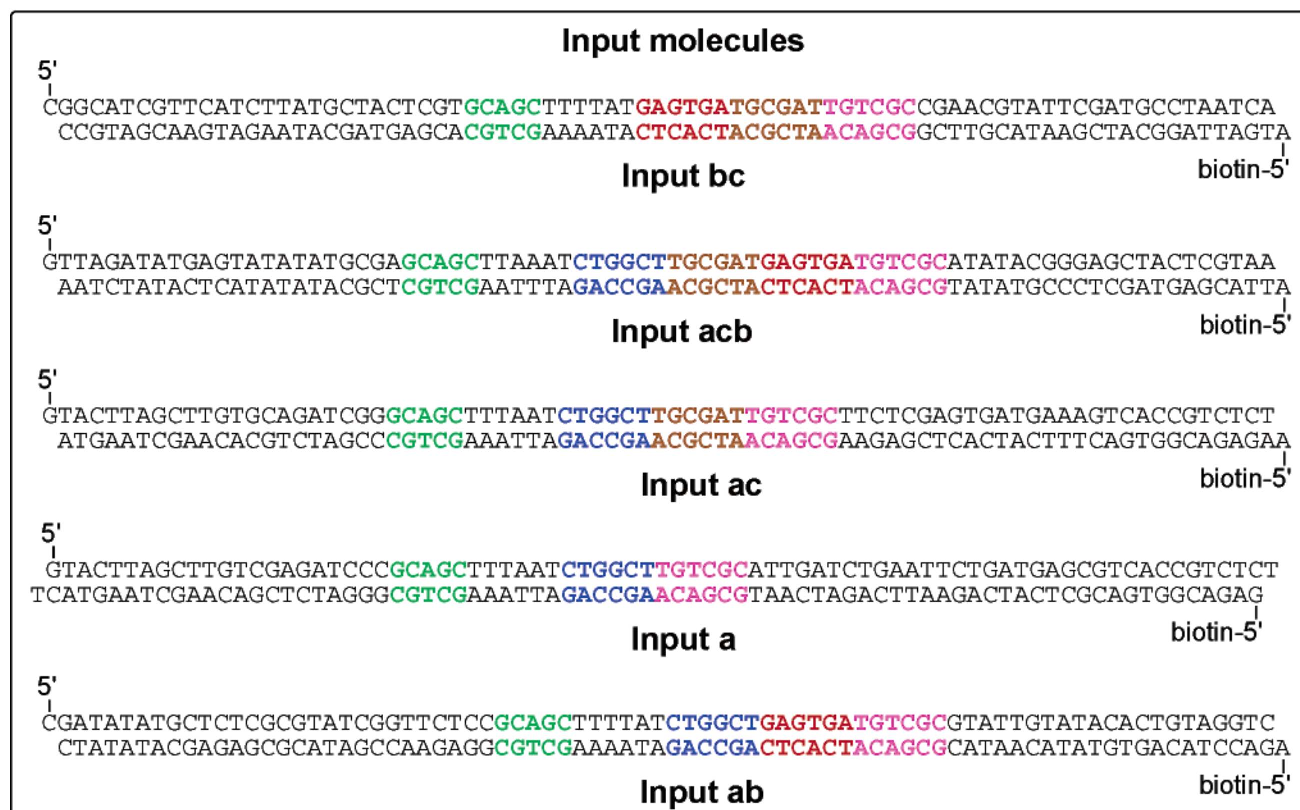


Figure 5. Representative input molecules, each containing a 5-bp *Bbv*I recognition site (green), various 6-bp symbols (blue, red, brown), and a 6-bp terminator (purple).

Figure 6. A set of nine transition molecules, each containing a *Bbv*I recognition site (green) and a unique 4-bp sticky end.

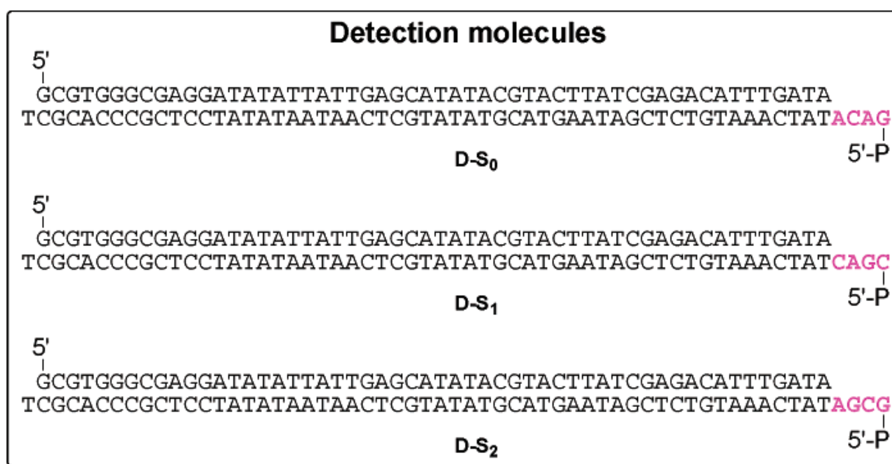


Figure 7. Three detection molecules, each containing a 4-bp sticky end (purple), which complement the three sticky ends produced by the various restriction modes of the terminator 6-bp domain.

in the Biacore sensorgrams. Furthermore, this stepwise technique minimized errors that could originate from nonspecific hybridization events. We examined the nonstepwise, one-pot option in one experiment with the immobilized input **ac**, where all

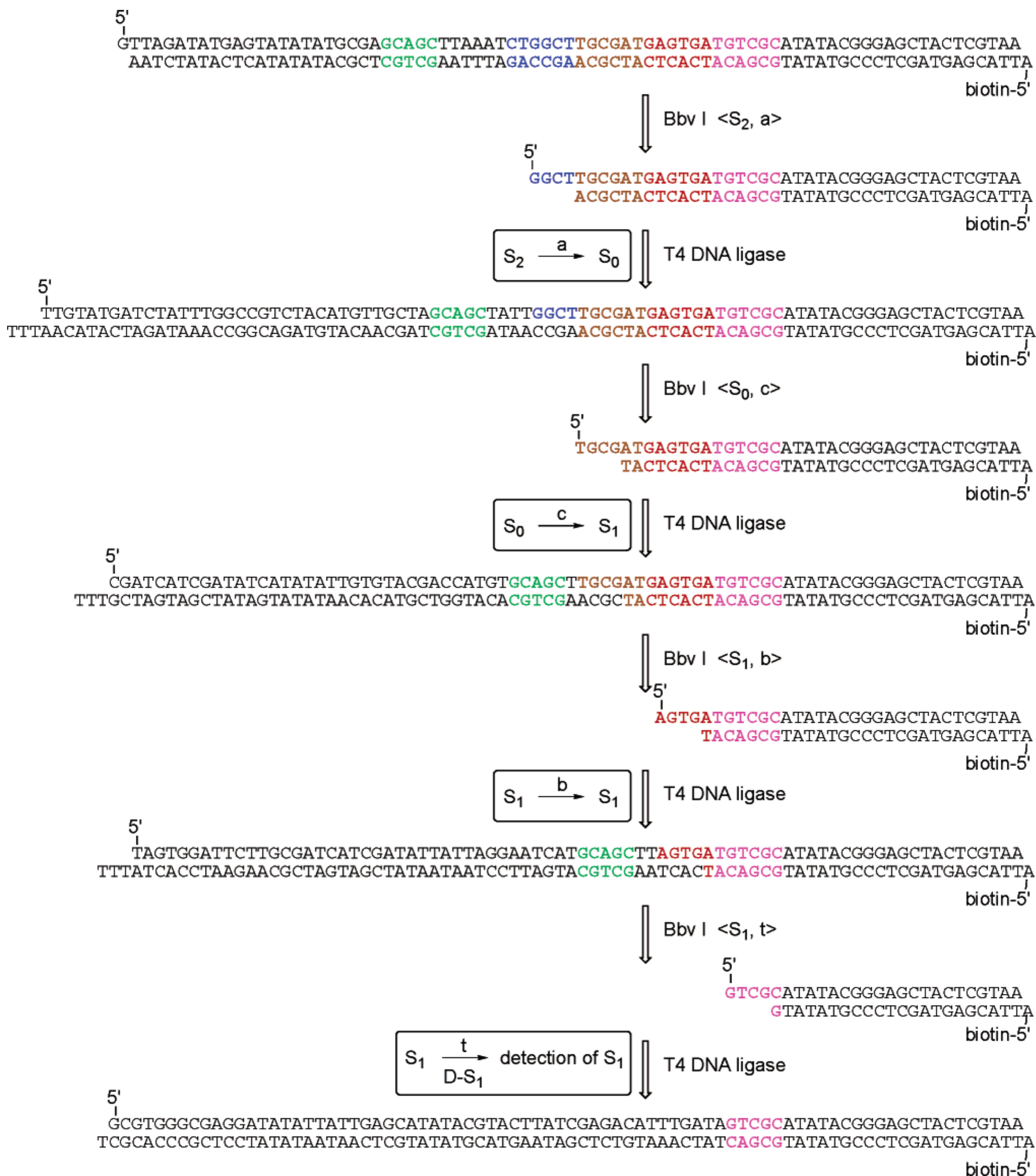


Figure 8. Computation with an input molecule, **acb**, starting with internal state S_2. The output is detected by specific hybridization with the relevant detection molecule, $D-S_1$.

soluble components were used in the same concentrations as was done in the stepwise protocol. Under these conditions, the computation failed to reach completion even after 2.5 h, as was concluded from the fact that no RU signal could be recorded upon injection of the detection mixture.

(10) (a) Frutos, A. G.; Liu, Q.; Thiel, A. J.; Sanner, A. M.; Condon, A. E.; Smith, L. M.; Corn, R. M. *Nucleic Acids Res.* **1997**, *25*, 4748−4757. (b) Frutos, A. G.; Smith, L. J.; Corn, R. M. *J. Am. Chem. Soc.* **1998**, *120*, 10277−10282.

(11) (a) Liu, Q.; Wang, L.; Frutos, A. G.; Condon, A. E.; Corn, R. M.; Smith, L. M. *Nature* **2000**, *403*, 175−179. (b) Wang, L.; Liu Q.; Corn, R. M.; Condon, A. E.; Smith, L. J. *J. Am. Chem. Soc.* **2000**, *122*, 7435−7440.

Accordingly, the experimental setup consisted of two solutions, one containing the restriction enzyme *Bbv*I, and the other containing a mixture of all transition molecules, T4 DNA ligase and ATP. Figure 8 illustrates the expected chemical events along the computation process with an **acb** input. We anticipated that the input molecule would be restricted by *Bbv*I to expose a four-nucleotide sticky end encoding for the initial state and the first input symbol. Next, the computation would proceed via ligation of the sticky end to the appropriate transition molecule. The product would then be restricted again by *Bbv*I at the next

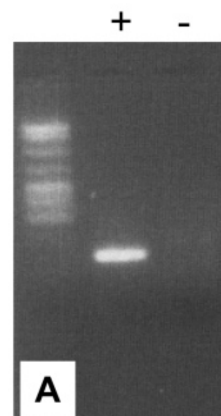
symbol to expose a new four-nucleotide sticky end. This process would continue in the same way along a series of ligations and restriction events. The computation would proceed until no transition molecule is available to match the exposed sticky end of the processed input, or until the terminator domain is restricted, forming a sticky end that encodes for the final state. In a step analogous to a print instruction of a conventional computer, this sticky end would ligate to one of the three output-detecting molecules. The resultant output reporter sequence would then be identified by the increased SPR response.

The new design of 3-symbol-3-state automata was first examined by experiments carried out in homogeneous solution using the automaton shown in Figure 4, in analogy to our previously described methods with 2-symbol-2-state automata.[1] Thus, computation with each of the input molecules, either **ab**, **a**, or **ac**, was done by mixing the input with all components, including *Bbv*I, T4 DNA ligase, ATP, the relevant transition molecules, and an appropriate detection molecule, either $D-S_2$, $D-S_0$, or $D-S_1$, respectively. Detection of the molecular output was carried out by PCR amplification followed by gel electrophoresis. The results confirmed that the expected output signals were indeed formed (Figure 9). No output signals could be detected in control experiments that lacked either the enzymes or the necessary primers for the PCR amplification.

The applicability of the SPR technology for output detection was tested by the use of immobilized detection molecules.[12] Three different sectors of a Biacore streptavidin chip were loaded with biotinylated detection molecules. The first sector was loaded with the S_0 detection molecule, while the second and third sectors were loaded with the S_2 detection molecule (Figure 10). Injection of a mixture containing S_2 output, DNA ligase, and ATP to the first and second sectors resulted in a positive response of the second sector with no observable change in the sensorgram of the first sector. Following this experiment, a solution containing the S_0 output was injected to all three sectors, leading to the expected positive response only of the first sector. These experiments have confirmed that the SPR technology can be used for output monitoring in real time with a high level of confidence.

Realization of our computational design was achieved using the automaton shown in Figure 4 as follows: each of the four sectors of the chip was loaded with a different biotinylated input. Computation was carried out by alternating the feed solution between the above-described restricting and ligating solutions. Switching from one solution to another, including an SDS wash between, was done automatically at predetermined time intervals. Thus, the flow cells were first fed with the solution of *Bbv*I, then washed with SDS, then fed with a mixture of the transition molecules and ligase, washed again with SDS, again fed with *Bbv*I, and so forth. The computation was terminated after executing a sufficient number of such cycles. Detection of the final state was carried out by a sequential feed of three detection mixtures, each containing T4 DNA ligase, ATP, and one of the detection molecules, $D-S_0$, $D-S_1$, or $D-S_2$.

To examine the above-described design of parallel computation with 3-symbol-3-states automata, we immobilized each of the four different inputs (**a**, **ac**, **acb**, and **bc**) on one of the four



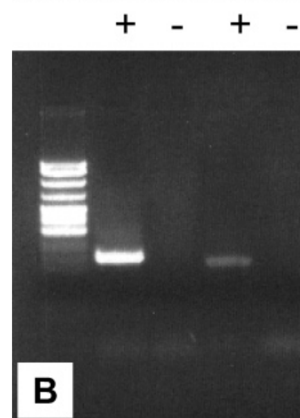


Figure 9. Gel electrophoresis monitoring of computations with 3-symbol-3-state automata in homogeneous solution. (A) Computation with input **a** followed by amplification of the output signal by PCR. The expected bands are observed only when the appropriate primers were used (+). Inappropriate primers were employed in the control experiments (−). (B) Control experiments with (+) and without (−) the hardware (*Bbv*I and ligase), using inputs **a** and **ac**.

sectors of a Biacore SA-chip. The increase of the corresponding RU values in the sensorgrams, 1385, 2376, 2379, and 2321 RU (Figure 11), reflected the corresponding input loading on each sector. In this experiment, we used only those transition molecules ($\mathbf{T_1}$, $\mathbf{T_2}$, and $\mathbf{T_3}$) that were actually needed to carry out the computation with inputs **a**, **ac**, and **acb**, but not those required for computation with **bc**. Expectedly, essentially no change in the sensorgram representing input **bc** could be observed during the computation process, while the other three inputs responded to the ligation and restriction cycles, as expected.

Following the proof of concept, we carried out a complete, parallel computation experiment, employing the entire set of nine transition molecules of the automaton (Figure 12). Accordingly, each one of the four inputs, **ab**, **a**, **ac**, and **acb**, was immobilized on a different sector of the SA chip, exhibiting increased sensorgram RU values of 2551, 1020, 1684, and 1662, respectively. Although the computation could be carried out automatically, all of the way to the detection of output signals, we performed the process stepwise to closely monitor each transformation. The restriction and ligation steps were reflected

(12) (a) Nilsson, P.; Persson, B.; Uhlen, M.; Nygren, P. A. *Anal. Biochem.* **1995**, *224*, 400−408. (b) Smith, E. A.; Kyo, M.; Kumasawa, H. K.; Nakatani, K.; Saito, I.; Corn, R. M. *J. Am. Chem. Soc.* **2002**, *124*, 6810−6811.

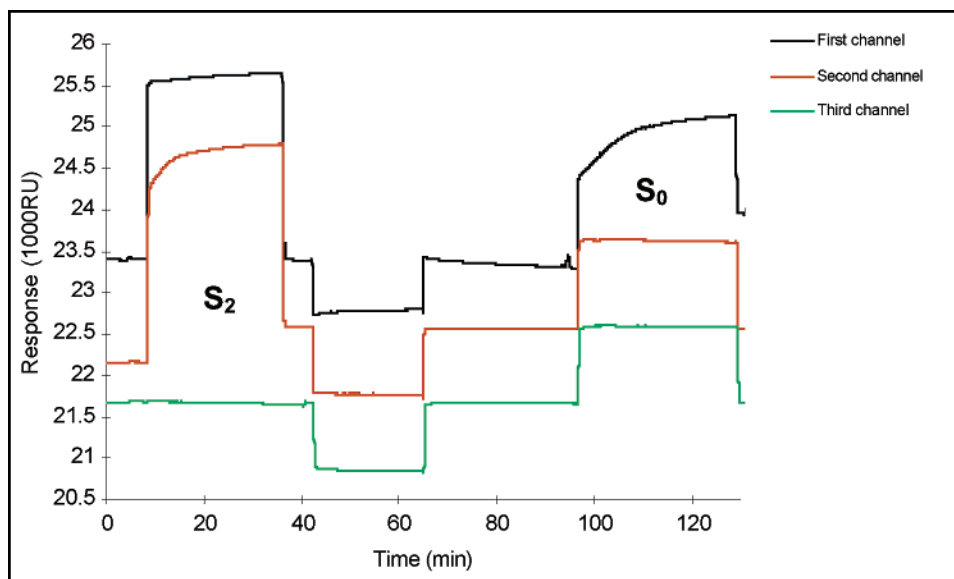


Figure 10. Output detection by immobilized detection molecules using SPR technology. The first sector of the Biacore SA-chip was loaded with the biotinylated S_0 detection molecule (black), while the second (orange) and third (green) sectors were loaded with the biotinylated S_2 detection molecule. Injection of an S_2 output resulted in positive response at the second sector. Injection of an S_0 output resulted in positive response at the first sector.

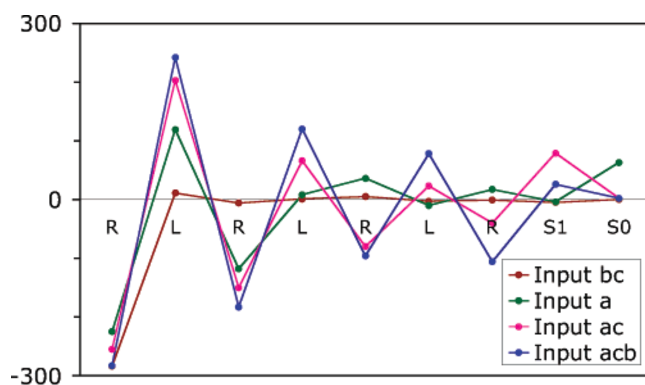


Figure 11. Stepwise parallel computation with four inputs: **bc**, **a**, **ac**, and **acb**. The transition molecules were supplied only to satisfy the computation needs of the latter three inputs, while no transition molecules were available for computation with the **bc** input. The differential RU values represent the changes in the SPR response between two consecutive steps. R represents restriction; L represents ligation. The computation was followed by detection with the detection molecules D−S_0 and D−S_1.

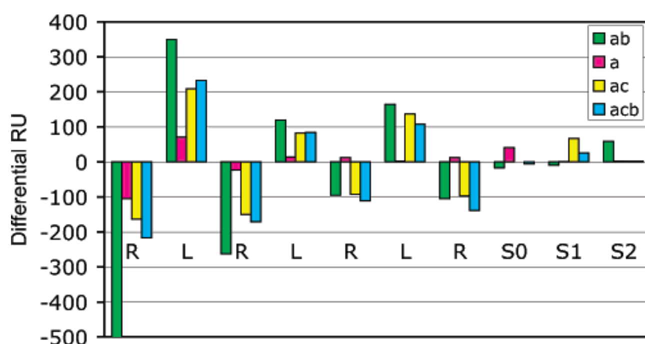


Figure 12. Stepwise parallel computation with four inputs: **ab**, **a**, **ac**, and **acb**. The differential RU values represent the changes in the SPR response between two consecutive steps. R represents restriction; L represents ligation. The output was detected by the positive response following injection of a solution containing one of the three detection molecules, D−S_0, D−S_1, or D−S_2.

by negative and positive changes in the SPR sensorgram, respectively. The computation was started by exposure of the input to the endonuclease, *Bbv*I, resulting in a negative change in all four channels. These changes were found to be proportional to the input load on each sector. Subsequent exposure to a mixture containing all nine transition molecules and ligase resulted in positive changes of the RU values (Figure 12). Although, under forcing conditions, each step could be pushed to completion, we found that allowing only 25−30 min for each step was sufficient to reach unequivocal detection of the output signal. Thus, because the longest input molecule consisted of three symbols, the system was allowed to progress along seven processing steps, starting with restriction and ending with restriction. This sequence of events was expected to expose different sticky ends that were produced upon restriction of the terminator domains, each immobilized on a different sector, ready for detection. The output detection was carried out through sequential injections of mixtures, each containing one of the three detection molecules, T4 DNA ligase and ATP. As seen in Figure 12, the injection of a detection mixture containing D−S_0 resulted in a significantly increased RU signal of the sector originally loaded with input **a**. Subsequent injection of the D−S_1 resulted in a positive response for two other sectors, originally loaded with inputs **ac** and **acb**. Finally, injection of D−S_2 resulted in the positive response of the fourth sector, originally loaded with input **ab**. These exclusive responses of each sector to the relevant detection mixture have unequivocally demonstrated the feasibility of the 3-symbol-3-state automata. The high fidelity of the system and the efficient detection allow for rapid processing of the restriction/ligation steps even with incomplete chemical yield in each step.

Experimental Section

General Methods. Deoxyoligonucleotides were custom-ordered (Sigma-Genosys), including 5′ phosphorylation and 5′ biotin modifications. Oligonucleotides (desalted, 0.05 μmol scale) were used without further purification. Concentrations obtained from the vendor were used to calculate molarity. Annealing was performed by heating a stoichiometric mixture of the complementary strands in One-Phor-All Buffer PLUS (Amersham Biosciences) (10 mM Tris-acetate, pH 7.5, 10 mM magnesium acetate, and 50 mM potassium acetate) to 95 °C followed by cooling to room temperature.

Surface Plasmon Resonance (SPR) Measurements. All measurements were performed on a Biacore 2000 and a Biacore 3000 (Biacore AB, Uppsala, Sweden). The inputs were immobilized on sensor chip surfaces precoated with streptavidin (Sensor chips SA, Biacore AB). The experiments were performed at a temperature of 25 °C. The flow rate was 2 μL/min. The running buffer was HBS (10 mM Hepes, pH 7.4, 0.3 M NaOH, 0.15 mM EDTA) supplemented with 0.005% surfactant P20 (Biacore AB). Prior to the experiment, the sensor chip was treated with short pulses of 50 mM sodium hydroxide, followed by 0.05% sodium dodecyl sulfate (SDS) pulse to precondition the chip surface.

Immobilization of Input Molecules. Each input molecule was dsDNA, 79-bp long, which was prepared by hybridization of a biotinylated oligonucleotide with its complementary sequence in One-Phor-All Buffer PLUS. Immobilization of the input molecule onto the surface was performed by injecting 2−20 μL of buffer solution containing the input (2.5 μM) and excess of the complementary nonbiotinylated strand (1 μM). Each input was injected solely into one flow channel of the chip.

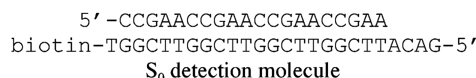
Stepwise Computation. The computation process was carried out by alternate injections of two solutions: (a) a 60 μL solution containing endonuclease *Bbv*I (0.03 u/μL, New England Biolabs) in One-Phor-All Buffer PLUS, and (b) a pool of transition molecules (dsDNA, 36−46 bp each, 2.5 μM of each) in One-Phor-All Buffer PLUS, supplemented with ATP (Sigma-Aldrich) and T4 DNA ligase (Promega) to final concentrations of 1 mM and 0.1 u/μL, respectively. Each computation cycle consisted of a simultaneous injection of solution a over all four different immobilized inputs, then injection of SDS (8 μL of 0.05%) to remove nonspecifically bound enzyme, and then injection of solution b (50 μL).

Output Detection. Detection was carried out by consecutive injections of solutions (40 μL each) of the detection molecules, 54-bp dsDNA, to all four channels at a flow rate of 2 μL/s. Each solution contained 5 μM detection output in One-Phor-All Buffer PLUS supplemented with ATP and T4 DNA ligase to final concentrations of 1 mM and 0.1 u/μL, respectively. Observation of a ligation signal in a flow channel identified the final state of the automaton in that specific channel.

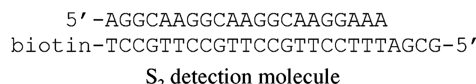
Computation in Homogeneous Solution. SPR results were supported by computation experiments carried out in solution, in analogy to the previously reported experiments.[1] The computation components were 0.02 μM of dsDNA input molecule, 0.15 μM of each of the nine transition molecules, 0.08 u/μL of *Bbv*I, and 0.15 u/μL of T4 DNA ligase, and were added to One-Phor-All Buffer PLUS, supplemented with 1 mM ATP and incubated at 37 °C for 120 min. The mixture also contained 0.035 μM of the matching detection molecule, as predicted from the computation program. Interaction between the produced output and the complementary detection molecule resulted in the formation of a dsDNA molecule. The presence of such a specific molecule was detected by PCR amplification and was assayed by gel electrophoresis using 4% MetaPhor agarose (FMC BioProducts). The lengths of the DNA species were verified using the molecular weight marker pB*R*322 DNA-Msp I Digest (New England Biolabs), and by way of comparison with synthetic primers.

Immobilized Detection Molecules. Detection molecules were immobilized on a SA Biacore chip in two steps. A 2.5 μM biotinylated ssDNA in buffer solution was injected (10−20 μL) into a single flow channel. Following this, the complementary strand in the buffer solution was injected (5 μM, 10−20 μL) and hybridized on the surface. The buffer used for the immobilization steps was HBS supplemented with 0.05% surfactant P20.

The bound dsDNA detection molecules (20−21 bp) that were formed were subjected to ligation experiments. Each ligation solution contained a dsDNA molecule (2.5 μM) with a complementary sticky end to one of the final states, in One-Phor-All Buffer PLUS supplemented with ATP and T4 DNA ligase to final concentrations of 1 mM and 0.1 u/μL, respectively. Due to an injection of a 40 μL ligation solution to the chip, a significant increase of an SPR signal in a specific flow channel was indicative of a ligation event. The oligonucleotides used as detection molecules were:



S_0 detection molecule



S_2 detection molecule

Polymerase Chain Reaction. A typical mixture (50 μL) for the PCR experiments included the computation solution (2 μL), PCR-D (50 ng/μL), and PCR-ac, PCR-a, or PCR-ab (50 ng/μL, 2 μL), using Taq polymerase. PCR program: step 1, 5 min at 94 °C; step 2, 2 min at 50 °C; step 3, 3 min at 72 °C; step 4, 45 s at 94 °C; step 5, 1 min at 50 °C; step 6, 3 min at 72 °C (steps 4−6 were repeated 28 times); step 7, 45 s at 94 °C; step 8, 90 s at 50 °C; step 9, 10 min at 72 °C; step 10, 1 h at 15 °C.

The oligonucleotides used for PCR amplification follow:

PCR-D, 5′-CGTGGGCGAGGATATATT; PCR-ac, 5′-GGTGAC-TTTCATCACTCGAG; PCR-a, 5′-GACGGTGACGCTCATCAG; and PCR-ab, 5′-GACCTACAGTGTATACAATACGC.

Conclusions

This work offers significant enhancement of the computational power of our previously reported 2-symbol-2-state automata via four major improvements. The first improvement is the increase of the number of internal states from 2 to 3 by applying three modes of restriction of each 6-bp symbol. The second improvement is the increase of the number of symbols from 2 to 3. These two improvements increase the overall number of syntactically distinct programs from 48 to 137 781. Theoretically, automata with up to 39 different symbols, 6 bp each, could be created on the basis of this technology, and in this work a set of 37 such symbols is reported.

The third improvement is the use of the SPR technology, which permits real-time detection of the output signal as well as real-time monitoring of stepwise computation. Furthermore, the immobilization of the input molecules on a Biacore chip enables the fourth improvement, which is the ability to perform parallel computation on different inputs. Beyond the direct benefit of simultaneous computation with a number of inputs, using the output of one set of computations as the input of another computation could increase the complexity of this parallel computation. These improvements would allow for more mathematically advanced computations as well as for new opportunities in the area of information encryption. For example, using a large number of immobilized input molecules, each serving as a pixel, could allow for encryption of visual images. Further conceptual improvements of these biomolecular computing devices,[13] as well as their applications, are currently being investigated in our laboratories.

Acknowledgment. This work was carried out at the Russell Berrie Nanotechnology Institute at the Technion. We thank the German-Israeli Project Cooperation (DIP) and the Skaggs Institute for Chemical Biology for their financial support. E. Keinan is an incumbent of the Benno Gitter & Ilana Ben-Ami chair of Biotechnology, Technion.

JA047168V

(13) Cavaliere, M.; Jonoska, N.; Yogev, S.; Piran, R.; Keinan, E.; Seeman, N. C. Proceedings of DNA10, Milano, Italy, June, 2004.