Solution of a Satisfiability Problem on a Gel-Based DNA Computer

Ravinderjit S. Braich, Cliff Johnson, Paul W.K. Rothemund,
Darryl Hwang, Nickolas Chelyapov, and Leonard M. Adleman

University of Southern California
Laboratory for Molecular Science
835 West 37th Street, SHS 172
Los Angeles, CA 90089

Abstract. We have succeeded in solving an instance of a 6-variable 11-clause 3-SAT problem on a gel-based DNA computer. Separations were performed using probes covalently bound to polyacrylamide gel. During the entire computation, DNA was retained within a single gel and moved via electrophoresis. The methods used appear to be readily automatable and should be suitable for problems of a significantly larger size.

1 Introduction

In 1994 Adleman demonstrated the use of DNA molecules as a means of solving computational problems [1]. The vast parallelism which computing with molecules potentially affords has led to speculation that molecular computers might be suitable for attacking problems that have resisted conventional methods [1, 2, 3, 4, 5, 6, 7, 8, 9]. While the theory of molecular computation has developed rapidly, the practice of molecular computation has not kept pace. Although several groups have investigated molecular computation in the laboratory [1, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20], no problem has yet been solved that most humans would find daunting. This paper reports on the progress of our group in attempting to create a molecular computer capable of solving problems that would be beyond the range of humans without the aid of electronic computers.
The model of computation described here is related to the previously described Sticker Model [21]. The Sticker Model uses two basic operations for computation: separation based on subsequence and application of stickers. In the experiment reported here, only separations were used.
Our initial approach to separation involved the incubation of a solution containing a DNA library with probes attached to a solid support (beads or filters). Molecules with appropriate subsequences hybridized to probes and were captured; molecules without such subsequences were removed by washing. Captured molecules were released back into solution by heating in new buffer. This seemingly straightforward approach did not work well in our hands. First, the hybridization of DNA in a 3-dimensional solution with probes immobilized on a 2-dimensional solid support was unacceptably slow. Second, molecules that should have been retained were lost at an unacceptably high rate during washing. Third, efficient release of captured molecules was achieved only with the use of a large volume of buffer, with the result that the DNA library became increasingly dilute as the computation progressed.
Recently, Mosaic Technologies (Boston, MA) introduced the Acrydite™ phosphoramidite for modifying DNA molecules at the $5'$-end during chemical synthesis. Like an acrylamide monomer, the Acrydite™ phosphoramidite carries a reactive ethylene functionality. Hence, standard methods can be used to copolymerize Acrydite™ -modified DNA probes into polyacrylamide gels—covalently linking the probes to the gel matrix. Because a gel is a nearly liquid environment, Acrydite™ -linked DNA probes apparently approximate probes in a solution. This gives one benefits of a solid-support-based system while still retaining characteristics of a solution-based system. In particular, the capture rate of molecules with the proper subsequence is improved, presumably because the 3-D to 2-D transition is mitigated. In addition, DNA can be moved by electrophoresis rather than transported by the movement of buffer; hence the problems of volume increase and library dilution are solved.
Our pilot separation experiments with Acrydite™ were adequate to warrant testing the technology in a DNA-based computation on a 3-SAT problem. We chose to solve the 6-variable 11-clause formula

$$\Phi = \begin{array}{llll} (x_1 \vee \neg x_2 \vee \neg x_3) & \wedge\ (x_2 \vee \neg x_3 \vee \neg x_4) & \wedge\ (x_3 \vee \neg x_4 \vee x_5) & \wedge \\ (x_4 \vee \neg x_5 \vee \neg x_6) & \wedge\ (x_5 \vee \neg x_6 \vee \neg x_1) & \wedge\ (x_6 \vee \neg x_1 \vee \neg x_2) & \wedge \\ (x_1 \vee x_2 \vee x_3) & \wedge\ (x_1 \vee x_2 \vee \neg x_3) & \wedge\ (\neg x_1 \vee x_2 \vee x_3) & \wedge \\ (\neg x_1 \vee x_2 \vee \neg x_3) & \wedge\ (x_1 \vee \neg x_2 \vee x_3) & & \end{array}. \quad (1)$$

Φ has a unique solution: $x_1 = x_2 = \cdots = x_6 = true$.

2 Materials and Methods

2.1 Design of the library

To represent all possible variable assignments for the chosen 6-variable SAT problem, a Lipton encoding [3] was used. For each of the 6 variables $x_1, x_2, \ldots, x_6$, two distinct 15 base *value sequences* were designed—one representing *true* (T), X_k^T, and one representing *false* (F), X_k^F. Each of the 2^6 truth assignments was represented by a *library sequence* of 90 bases consisting of the concatenation of one value sequence for each variable. DNA molecules with library sequences are termed *library strands* and a combinatorial pool containing library strands is termed a *library*. The probes used for separating the library strands have sequences complementary to the value sequences.

Errors in the separation of the library strands are errors in the computation. Sequences must be designed to ensure that library strands have little secondary structure which might inhibit intended probe-library hybridization. The design must also exclude sequences that might encourage unintended probe-library hybridization. To help achieve these goals, sequences were computer-generated to satisfy the following constraints:

1. Library sequences contain only A's, T's, and C's.
2. All library and probe sequences have no occurrence of 5 or more consecutive identical nucleotides; i.e. no runs of more than 4 A's, 4 T's, 4 C's or 4 G's occur in any library or probe sequences.
3. Every probe sequence has at least 4 mismatches with all 15 base alignment of any library sequence (except for with its matching value sequence).
4. Every 15 base subsequence of a library sequence has at least 4 mismatches with all 15 base alignment of itself or any other library sequence.
5. No probe sequence has a run of more than 7 matches with any 8 base alignment of any library sequence (except for with its matching value sequence).
6. No library sequence has a run of more than 7 matches with any 8 base alignment of itself or any other library sequence.
7. Every probe sequence has 4, 5, or 6 Gs in its sequence.

Constraint (1) is motivated by the assumption that library strands composed only of As, Ts, and Cs will have less secondary structure than those composed of As, Ts, Cs, and Gs [22]. Constraint (2) is motivated by two assumptions: first, that long homopolymer tracts may have unusual secondary structure and second, that the melting temperatures of probe-library hybrids will be more uniform if none of the probe-library hybrids involve long homopolymer tracts. Constraints (3) and (5) are intended to ensure that probes bind only weakly where they are not intended to bind. Constraints (4) and (6) are intended to ensure that library strands have a low affinity for themselves. Constraint (7) is intended to ensure that intended probe-library pairings have uniform melting temperatures.

The value sequences generated to represent $x_1 = F, x_2 = F, \cdots, x_6 = F$ were:

$$\begin{array}{ll} X_1^F = 5' - \texttt{TATTCTCACCCATAA} - 3', & X_2^F = 5' - \texttt{ACACTATCAACATCA} - 3' \\ X_3^F = 5' - \texttt{CCTTTACCTCAATAA} - 3', & X_4^F = 5' - \texttt{CTCCCAAATAACATT} - 3' \\ X_5^F = 5' - \texttt{AACTTCACCCCTATA} - 3', & X_6^F = 5' - \texttt{TCATATCAACTCCAC} - 3' \end{array}$$

The value sequences generated to represent $x_1 = T, x_2 = T, \cdots, x_6 = T$ were:

$$\begin{array}{ll} X_1^T = 5' - \texttt{CTATTTATATCCACC} - 3', & X_2^T = 5' - \texttt{ACACCTAACTAAACT} - 3' \\ X_3^T = 5' - \texttt{CTACCCTATTCTACT} - 3', & X_4^T = 5' - \texttt{ATCTTTAAATACCCC} - 3' \\ X_5^T = 5' - \texttt{TCCATTTCTCCATAT} - 3', & X_6^T = 5' - \texttt{TTTCTTCCATCACAT} - 3' \end{array}$$

We note that because of the nature of the constraints (which require the inspection of subsequences ≤ 15 bases long) it was only necessary to check that a special subset of all 2^6 library sequences satisfied the constraints. In particular, the sequence design program checked that the library sequences $X_1^T X_2^T X_3^T X_4^T X_5^T X_6^T$, $X_1^F X_2^F X_3^F X_4^F X_5^F X_6^F$, $X_1^F X_2^T X_3^F X_4^T X_5^F X_6^T$, and $X_1^T X_2^F X_3^T X_4^F X_5^T X_6^F$ simultaneously satisfied constraints (1–7). These sequences contain, as subsequences, all 15 base subsequences that occur in the full 64 sequence set of library sequences. Thus, for longer library sequences, the number of constraints that need to be checked does not increase exponentially with the number of variables but rather as the square of the number of variables. We denote the reverse complements of X_k^T and X_k^F as $\overline{X}_k^T$ and $\overline{X}_k^F$, respectively. We sometimes refer to "X_k^T or X_k^F" and "$\overline{X}_k^T$ or $\overline{X}_k^F$" as X_k and $\overline{X}_k$, respectively.

2.2 Synthesis of the library and probes

The 6-variable library strands were synthesized by employing a mix-and-split combinatorial synthesis technique [23]. Oligonucleotide synthesis was performed on a dual column ABI 392 DNA/RNA Synthesizer (Applied Biosystems, Foster City, CA) at a 1μmole scale on CPG solid support. The library strands were assigned library sequences with X_1 at the $5'$ end and X_6 at the $3'$-end ($5' - X_1 - X_2 - X_3 - X_4 - X_5 - X_6 - 3'$). Thus synthesis began by assembling the two 15 base oligonucleotides with sequences X_6^T and X_6^F in separate columns. The columns were then removed from the synthesizer and opened; the CPG beads in the columns were removed and mixed together. One half of the beads were returned to the first column and the other half to the second. Synthesis continued with sequences X_5^T and X_5^F. This process was repeated until all 6 variables had been treated.
Twelve probes, having sequences $\overline{X}_k^F$, $\overline{X}_k^T$, $k = 1 \ldots 6$ and modified at the $5'$-end with Acrydite™ , were obtained from Operon Technologies Inc. (Alameda, CA).

2.3 Library capture analysis

To determine the efficiency of library capture and release by gel-embedded probes, capture experiments were undertaken. In this experiment, a library similar to that described in 2.2 was used but the synthesis was performed using a polystyrene rather than a CPG support.

Preparation of gels. Capture gels were prepared in 1 mm x 10 cm x 10 cm plastic gel cassettes (Novex). The upper half of the gel cassette was divided into three parts by inserting ~1-mm thick, ~5-mm wide, plastic spacers. Approximately 7 ml of 10% acrylamide solution were poured into the cassette, enough to cover the bottom 0.5 cm of the dividing spacers, and allowed to polymerize. After the acrylamide had polymerized for 10 minutes, any unpolymerized solution was shaken off, and the top of the polymerized gel rinsed with 1X TBE buffer. Capture layers were then polymerized on top of the already polymerized gel. In each partition of the gel, 100 μl of 10% acrylamide solution containing 15μM of the appropriate probe were allowed to polymerize. Again polymerization was allowed to proceed for 10 minutes, excess solution shaken off, and top of the gel layer rinsed with 1X TBE. At this point, dividing spacers were removed and top of the gel rinsed one more time with 1X TBE. The gel was then topped off with 10% acrylamide solution and appropriate combs for well formation inserted. Two sets of capture gels were prepared and used in the library capture experiment.

Running the gels. For one set of gels, electrophoresis was carried out in the fridge (cold) at 4°C in order to observe capture. For the other set of gels, electrophoresis was carried out using a gel box with a water circulator (hot) set to 75°C. In both cases gels were put in the electrophoresis chamber and allowed to come to thermal equilibrium before commencing electrophoresis. Electrophoresis was carried out at 10 volt/cm^2 in all cases. It was observed that in the higher temperature electrophoresis went at a faster rate than when electrophoresis was carried out in the cold. After electrophoresis was complete, gels were dried on a gel dryer for 30 minutes at 40°C. After drying, the gels were put in Phosphor Storage Screens and exposed overnight.

2.4 Confirming integrity of the library via PCR

To verify the degeneracy and integrity of the library, the library was amplified via PCR. Twenty PCR reactions were performed on the library using $5'$-end primers with sequences X_1^T or X_1^F and $3'$-end primers with sequences $\overline{X}_2^T, \ldots, \overline{X}_6^T$ or $\overline{X}_2^F, \ldots, \overline{X}_6^F$.

2.5 The algorithm

Coupling of the Acrydite™ phosphoramidite to DNA probes allows the probes to be immobilized in a polyacrylamide gel matrix. During electrophoresis at low temperatures, such probes hybridize with and capture passing DNA molecules bearing complementary subsequences. DNA molecules without complementary subsequences pass through the gel relatively unhindered. Captured DNA strands can be released by running electrophoresis at a temperature higher than the melting temperature of a probe/probe-complement hybrid. Released molecules can be used in subsequent steps as required. Using this approach, our algorithm is as follows:

1. For each of the 11 clauses of Φ prepare a polyacrylamide gel *capture layer* containing three Acrydite™ -modified probes, one for each literal in the clause. (If x_k appears in the clause, add a probe with sequence $\overline{X}_k^T$; if $\neg x_k$ appears add a probe with sequence for $\overline{X}_k^F$.) Place the capture layers in sequence within a single gel. Place the library into the gel preceding the first capture layer.
2. Cool the area of the gel containing the first capture layer while heating the areas of the gel preceding and following it. Begin electrophoresis to move the library through the first capture layer. Molecules encoding truth assignments satisfying the first clause will be captured in the first capture layer, while molecules encoding non-satisfying assignments will run through the first capture layer and continue beyond the second capture layer.
3. Cool the area of the gel containing the second capture layer while heating the areas of the gel preceding and following it. Molecules captured in the first capture layer will be released to move through the second capture layer. Released molecules encoding truth assignments satisfying the second clause will be captured in the second capture layer, while molecules encoding non-satisfying assignments will run through the second capture layer and continue beyond the third capture layer.

This process is repeated for each of the remaining 9 capture layers. The final (11th) capture layer will capture only those molecules which have been retained in all 11 capture layers and hence encode truth assignments satisfying each clause of Φ. These *answer strands* are extracted from the final capture layer, PCR-amplified and sequenced.

2.6 Construction and running of the computer

Preparation of the modules. For each clause in Φ a *clause module* (Fig. 1B) containing a 3-probe capture layer was prepared. The capture layer was prepared by mixing the three probes (chosen as described in step 1 of the algorithm above) at a concentration of 7.5 μM each in 10% acrylamide solution. 100 μl of this solution were allowed to polymerize on top of 200 μl of already-solidified 0.7% (w/v) agarose (Seakem® Gold, FMC BioProducts, Rockland, ME) in a well of a 96-well flat bottom plate (Nunc-Immuno Plate, Nalge Nunc, Rochester, NY). Once the acrylamide layer had polymerized (10-15 minutes), warm agarose solution was added to fill the remainder of the well and allowed to solidify. Using this method, 11 clause modules were prepared. In addition, a *library module* was prepared by mixing 500 pmols of the library with agarose and allowing the agarose to solidify in one of the wells. Other *blank modules* were prepared by allowing pure agarose to solidify in some wells.

Loading the modules. The computation was performed in a 35-cm long glass tube with an outer diameter of 0.5 cm and an inner diameter of 0.3 cm. Before loading the tube with modules, the inside of the tube was silanized with Sigmacote® (Aldrich, Milwaukee, WI). The tube was then loaded with modules by a squishing method (Fig. 1C–E) wherein the tube was pushed into the appropriate well. This process cut a cylindrical core of gel from the well, transferred it into the tube, and forced resident modules upward. To ensure good contact between the successive modules and to reduce the possibility of air bubble formation at the interface, prior to each squish, resident modules were pushed downward until a small bit of the lowest module protruded from the tube. First several blank modules were added to the tube, followed by alternating clause modules and blank modules. Thus each capture module was separated from the next by a blank module of pure agarose gel. After all of the 11 clause modules had been loaded in the tube, a blank module and then finally the library module were loaded.

Heating and cooling the capture layers. To keep the temperature high or low at a given position on the glass tube, three movable water jackets were assembled by drilling 0.5-cm holes in plastic drying tubes (Aldrich). These water jackets were connected to hot or cold water circulators with plastic tubing and slid onto the glass tube. Figure 2 shows a schematic of the final apparatus.

2.7 Computation

The ends of the assembled apparatus were inserted into capped glass Liquid Scintillation Vials (Wheaton, Millville, N.J.) containing, 1X TBE gel running buffer and electrodes. The water jackets were adjusted so that the cold water jacket was positioned over the first capture layer. Throughout the computation, the hot circulating water bath was set at 75°C while the cold bath was maintained at 4°C. Gel electrophoresis was performed at a constant voltage of 307 volts, ~3 mA. After 30 minutes had passed, the electrophoresis was stopped, each water jacket was moved to the next capture layer, and the electrophoresis was restarted. This process was repeated for all 11 capture layers.

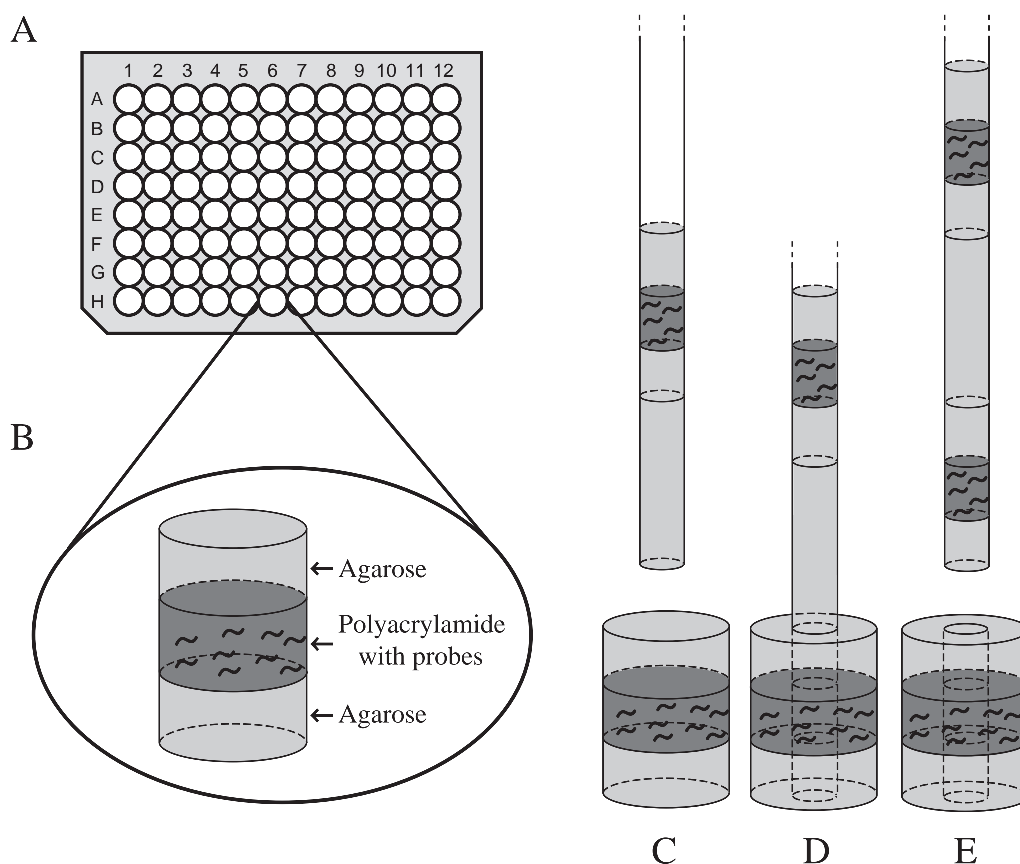

Fig. 1. Preparation of a clause module. (A) A 96-well flat bottom plate was used as a mold for the clause modules (as well as the library and blank modules). (B) To make a clause module, a polyacrylamide capture layer (with appropriate probes) was poured on top of already-solidified agarose gel. The well was then topped-off with agarose. (C–E) Loading of a clause module. (C) A glass tube holding one clause module and one blank module is positioned over a well holding a second clause module. (D) The glass tube is lowered into the well and the resident modules are pushed upwards. (E) The tube, now holding two clause modules and one blank module, is withdrawn.

Next the gel was extruded from the glass tube and the final capture layer was dissected away, crushed and soaked in 5 ml of water. The captured answer strands were extracted from the gel by incubating at 65°C for 12 hours. After extraction, the DNA was lyophilized and reconstituted into 200 μl of water. The DNA was desalted using a Microcon 30 Microconcentrator column (Millipore, Bedford, MA) and reconstituted in 500 μl of water.

2.8 Determination of answer strand

PCR. PCR amplification of the answer strands was performed on a PE Applied Biosystems GeneAmp® PCR System 9700 (Perkin Elmer, Foster City, CA). Five PCR reactions were run. For the first four reactions oligos with sequences X_1^T or X_1^F were used as 5′-end primers and Acrydite™ -modified probes with sequences $\overline{X}_6^T$ or $\overline{X}_6^F$ were used as 3′-end primers. In the fifth PCR reaction all four primers were used simultaneously. The PCR reactions were performed using $\frac{1}{5000}$th (by volume) of the reconstituted answer strands and 10 pmols each of the appropriate primers in a final reaction volume of 50 μl that contained 50 mM KCl, 1.5 mM $MgCl_2$, 10mM tris (pH 8.8), 200 μM of each dNTP, and 1 unit of Taq DNA Polymerase (Promega, WI). The reaction mixture was preheated to 95°C and thermocycled (95°C 15s, 40°C, 45s, 72°C, 90s) 35 times. To determine the number

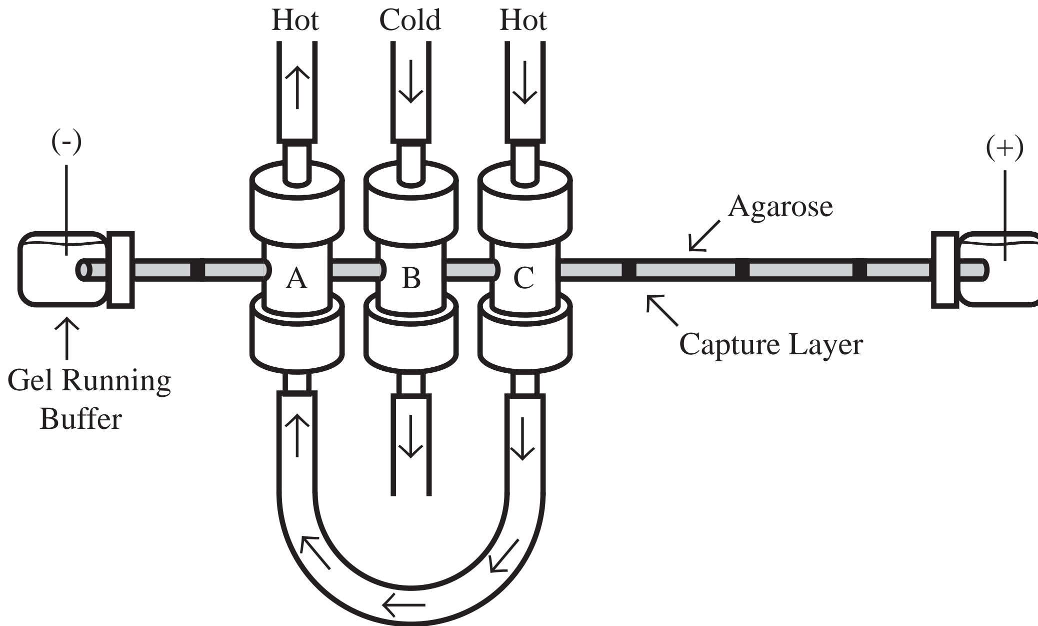

Fig. 2. Apparatus assembled for computation. A 35-cm glass tube loaded with the library module, 11 blank modules, and 11 clause modules was fitted with three water jackets (A-C). Library strands in the capture layer inside of (A) were released and moved into the capture layer inside of (B). There, library strands with subsequences complementary to the probes were captured and retained. The rest of the strands passed into the capture layer inside of (C) but because (C) was kept hot the strands passed through unhindered.

of correct answer strands recovered, additional PCR reactions (using all primer pairs and all four primers) were performed using fractions of the recovered answer strands from $\frac{1}{500}$th down to $\frac{1}{5\times10^{13}}$th.

Sequencing. Two sequencing reactions were run. The product of PCR amplification using primers with sequences X_1^T and $\overline{X}_6^T$ was sequenced using a primer with sequence X_1^T. In addition, the product of PCR amplification using all four primers ($X_1^T, X_1^F, \overline{X}_6^T$, and $\overline{X}_6^F$) was sequenced using a primer corresponding to X_1^F.
Prior to sequencing, 5 μl of the PCR product was incubated for 15 minutes at 37°C with 1 μl (2 units) of Shrimp Alkaline Phosphatase and 1 μl (10 units) of Exonuclease I (PCR Product Pre-Sequencing Kit, USB, Cleveland, OH). This pretreatment was performed to destroy any dNTP's, primers and extraneous single-stranded molecules left over from the PCR reaction that might have interfered with the sequencing reaction. After incubation the reaction tube was heated to 80°C for 15 minutes to inactivate the enzymes.
Pretreated PCR product was sequenced using the Thermo Sequenase Radiolabeled Terminator Cycle Sequencing Kit (USB). The sequencing reaction was run through 30 thermocycles (denatured at 95°C for 15s, annealed at 40°C for 45s, and extended at 72°C for 90s) on a GeneMate Thermocycler (ISC Bioexpress, Kaysville, UT).

3 Results

3.1 Library capture analysis

Figure 3 shows the capture of the library using each of the twelve possible probes. At low temperature library was captured on each of the probes. This confirmed both that library strands with subsequences corresponding to each value sequence were present in the library and that the probes were good (sometimes incompletely modified

probes failed to copolymerize into gels). At high temperature, library passed the probes unhindered suggesting that library strands could be efficiently released from probes at each step in the computation.

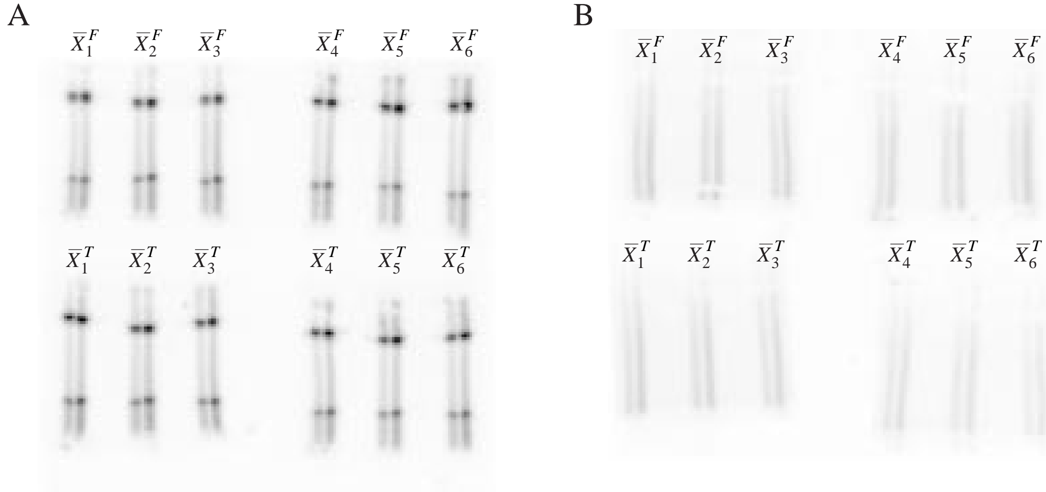

Fig. 3. Capture of the library by gel-embedded probes. (A) Twelve probes $\overline{X}_1^F, \dots, \overline{X}_6^F$ and $\overline{X}_1^T, \dots, \overline{X}_6^T$ were used to assay the capture of library strands bearing sequences $X_1^F, \dots, X_6^F$ and $X_1^F, \dots, X_6^F$ at low temperature. Upper bands show capture of the library on probes, lower bands show uncaptured library that presumably does not bear a subsequence complementary to the probe. (B) Repetition of the experiment at high temperature shows that library strands passed the probes unhindered (and hence could be released from a capture layer) at high temperature.

3.2 Confirming the integrity of the library via PCR

Figure 4 shows the results of the PCR reactions. PCR products of the expected lengths were obtained confirming that library strands with the correct subsequences corresponding to each value sequence (*true* or *false*) were present and in the expected positions in the library.

3.3 Readout of the answer strands by PCR

Figure 5 shows the results of the PCR amplification of $\frac{1}{5000}$th of the answer strands using all 4 combinations of the primers. When primers with sequences X_1^T and $\overline{X}_6^T$ were used, a 90-mer PCR product was seen. For the other 3 combinations of primers, little if any amplification was seen. Amplification of the original library was seen to give a 90-mer PCR product with each of the 4 different combinations of the primers. This indicates that the answer strands were enriched for strands encoding $x_1 = T$ and $x_6 = T$. Additional PCR reactions at other dilutions revealed that incorrect strands are present in small numbers (PCR of $\frac{1}{500}$th of the answer strands gave positive bands for all pairs of primers) and that the correct strands are present in great numbers (PCR of $\frac{1}{5\times10^{11}}$th still gave a positive band for X_1^T and $\overline{X}_6^T$ primers). Assuming that PCR allows the detection of single molecules, these PCR results allow us to approximate the number of correct strands and incorrect strands present in the recovered answer strands. Assuming that strands amplified by the primers X_1^T and $\overline{X}_6^T$ are correct strands, at least 5×10^{11} correct strands were present in the answer strands. Given that 500 pmols (3.0×10^{14} strands) were input to the computation, $\frac{1}{64}$th of these, or 4.7×10^{12} correct strands were input to the computation. Thus approximately 11% of the correct strands were retained at the end of the computation. Assuming that those strands amplified by the primers X_1^T and $\overline{X}_6^F$ were a single type of incorrect strand (bearing X_6^F and X_k^T for $k = 1 \dots 5$) there were less than 5000 of such strands. Assuming that all types of incorrect strands are present with this frequency

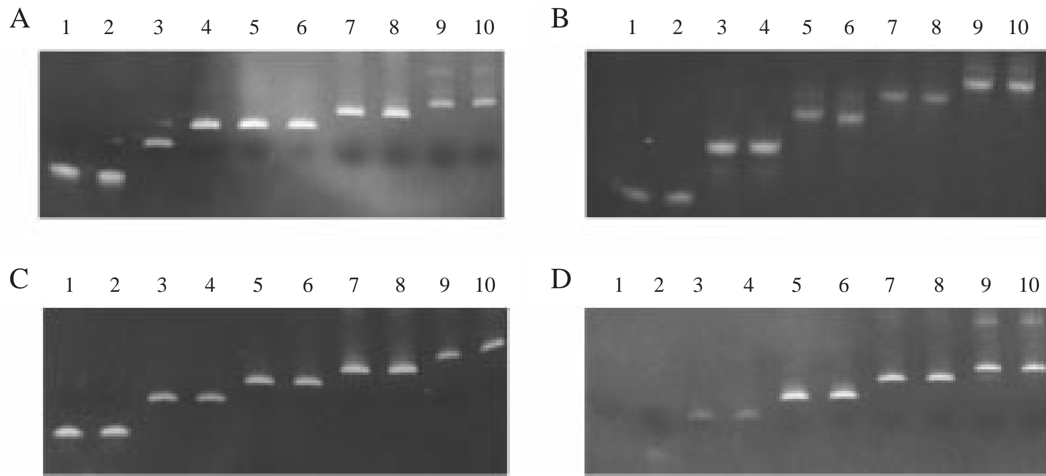

Fig. 4. PCR analysis of the original library. (A) X_1^T, $\overline{X}_2^T, \ldots, \overline{X}_6^T$ probes. (B) X_1^F, $\overline{X}_2^T, \ldots, \overline{X}_6^T$ probes. (C) X_1^T, $\overline{X}_2^F, \ldots, \overline{X}_6^F$ probes. (D) X_2^F and $\overline{X}_2^F, \ldots, \overline{X}_6^F$ probes. In each panel all lanes have X_1 as one primer and have as the other primer: lanes 2 and 3 $\overline{X}_2$ probe primer; lanes 3 and 4, $\overline{X}_3$ primer except for in panel (A) where only lane 3 has the $\overline{X}_3$ probe primer while lane has $\overline{X}_4$ probe primer; lanes 5 and 6, $\overline{X}_4$ primer, but see above for panel (A); lanes 7 and 8, $\overline{X}_5$ primer; and lanes 9 and 10, $\overline{X}_6$ primer.

means that at most 315,000 incorrect strands were present in the answer strands. This suggests correct strands outnumbered incorrect strands by a factor of at least 1.6 million, an enrichment from their original proportions by a factor of 100 million.

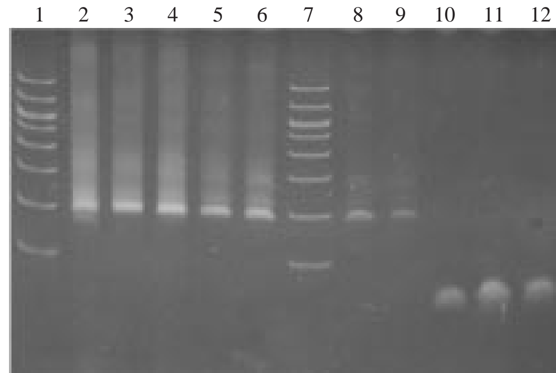

Fig. 5. Readout of the answer by PCR. Lanes 1 and 7, molecular size marker ladder. Lanes 2 - 6, library. Lanes 8 - 12, diluted answer strands. Lanes 2 and 8, all four primers. Lanes 3 and 9, X_1^T and $\overline{X}_6^T$ probe. Lanes 4 and 10, X_1^T and $\overline{X}_6^F$ probe. Lanes 5 and 11, X_1^F and $\overline{X}_6^T$ probe. Lanes 6 and 10, X_1^F and $\overline{X}_6^F$ probe.

3.4 Sequencing of the answer strands

Figure 6A and B show the results of sequencing the amplified answer strands using a primer with sequence X_1^T. Figure 6 shows that there is no degeneracy at any position, indicating that a unique computational solution was obtained. The unique solution corresponded to $x_2 = T, x_3 = T \ldots, x_6 = T$. Since $x_1 = T$ and $x_6 = T$ had already been established in the PCR step, it can be concluded that the answer strands correspond to $x_1 = T$, $x_2 = T$, $x_3 = T$, $x_4 = T$, $x_5 = T$, and $x_6 = T$, indicating a successful computation.

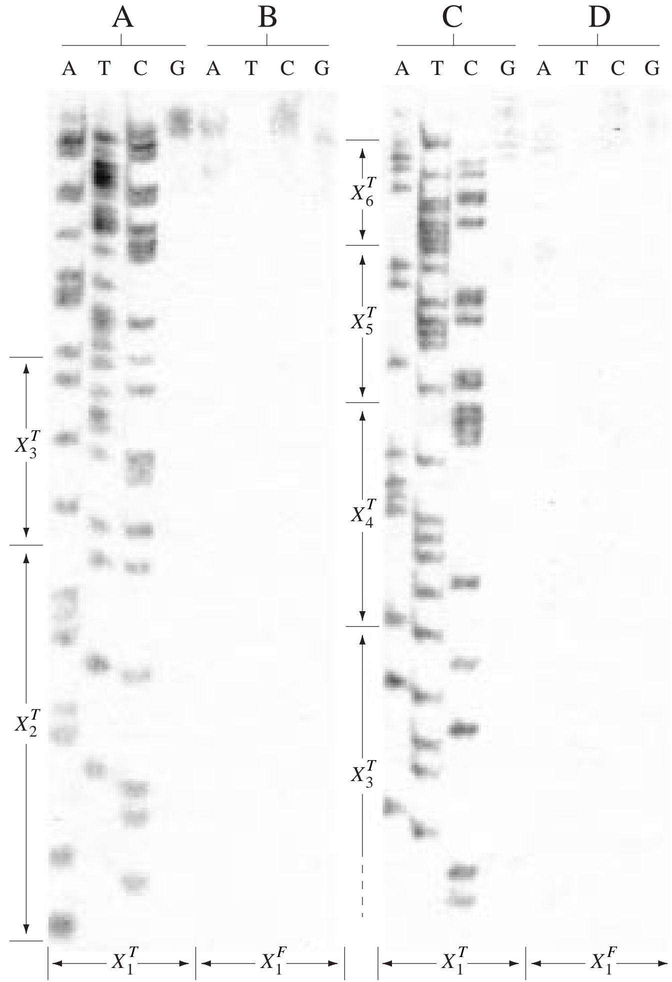

Fig. 6. Sequencing of the diluted answer strands. Termination nucleotides are shown at the top of each lane. The primer used in each sequencing reaction is indicated at the bottom of the lanes. Lanes (A) and (C) show sequencing of the PCR amplification of answer strands using primers with sequences X_1^T and $\overline{X}_6^T$ (lane 9 from Fig. 5). Lanes (A) shows that the strands contain, as subsequences, the sequences X_2^T and X_3^T while lanes (B) show the answer strands contain, as subsequences, part of X_3^T and all of $X_4^T, \ldots, X_6^T$. Lanes (B) and (D) show sequencing of the PCR amplification of answer strands using all four primers (lane 8 from Fig. 5). The absence of sequencing product in these lanes indicates that no strands representing any assignment including $x_1 = F$ were present in the answer strands.

4 Prospects for scaling up

Whether SAT problems of greater size may be solved depends on the difficulty of scaling up each of three procedures: design of the library strands, synthesis of the library strands, and execution of the computation. As for the first procedure, we note that the sequences $X_1^T, \ldots, X_6^T$ and $X_1^F, \ldots, X_6^F$ are a subset of 72 sequences designed for a larger 36-variable SAT problem and that sequences for 50-variable SAT problems with the same constraints have been designed. Assuming that longer library strands composed of these sequences perform as well as their shorter variants, sequence design does not seem to be a limiting factor for the solution of SAT problems with up to ten times as many variables as that solved here.
As for the second procedure, we plan to synthesize library strands for a 20-variable SAT problem via the ligation of three pools of shorter library strands: two pools of 105-base long 7-variable library strands and one pool of 90-base long 6-variable library strands. The 6-variable and 7-variable library strands have already been synthesized by a mix-and-split synthesis. Each library is being tested separately by running a capture analysis and simple computation as described for the 6-variable library. We have just begun experiments to ligate these pools into a full 20-variable library.
As for the third procedure, we believe that the results of our 6-variable computation show that our ability to capture, release, and recover correct answer strands is good enough to complete a successful 20-variable computation. Consider that after capture in and release from 11 clause modules $\sim 11\%$ of the correct answer strands were recovered. This suggests that in each module approximately 82% of the correct answer strands were captured. In an analogous 20-variable computation, starting with 500 pmols of library strands, there would be roughly 300 million correct answer strands. After passing 25 clause modules (for the analogous SAT formula) approximately 2 million correct answer strands would remain—enough to be easily detected by PCR.

5 Discussion

We have carried out a successful DNA computation on a 6-variable SAT problem. The correct solution was culled from 64 alternatives. This is slightly smaller than the number of alternative solutions (512) recently handled at Princeton by Faulhammer et al. [16] and slightly more than the number (16) handled the University of Wisconsin-Madison by Liu et al. [20] By solving small computational problems, these experimentalists and others have demonstrated the viability of several different architectures for DNA computing. It seems clear that the next objective should be the solution of problems which are beyond the capabilities of humans without the aid of electronic computers. Our success with a 6-variable 11-clause 3-SAT problem fortifies our view that we now possess the tools necessary to carry out such a computation. We are currently in the process of synthesizing a 20-bit library in order to solve a 20-variable SAT problem in the near future. We are also optimistic about the prospects of building an automated device for carrying out such computations. Despite our optimism, we must still acknowledge that the road to a DNA computer capable of solving computational problems which cannot be solved by electronic computers is a difficult one. In our opinion, creation of such a molecular computer will not be accomplished by incremental improvements in current approaches—breakthroughs will be needed.

6 Acknowledgements

This work was supported by grants from the National Aeronautics and Space Administration/Jet Propulsion Laboratory, the Defense Advanced Research Projects Agency, the Office of Naval Research, and the National Science Foundation.

References

[1] Leonard M. Adleman. Molecular computation of solutions to combinatorial problems. *Science*, 266:1021–1024, November 11, 1994.

[2] Richard J. Lipton. DNA solution of hard computational problems. *Science*, 218:17–26, 1996.

[3] Dan Boneh, Christopher Dunworth, and Richard J. Lipton. Breaking DES using a molecular computer. In Richard J. Lipton and Eric B. Baum, editors, *DNA Based Computers: Proceedings of a DIMACS Workshop, April 4, 1995, Princeton University*, volume 27 of *DIMACS: Series in Discrete Mathematics and Theoretical Computer Science*, pages 37–65, Providence, RI, 1996. American Mathematical Society.

[4] Leonard M. Adleman, Paul W. K. Rothemund, Sam Roweis, and Erik Winfree. On applying molecular computation to the data encryption standard. *Journal of Computational Biology*, 6(1):53–63, 1999.

[5] N. Jonoska and S. A. Karl. A molecular computation of the road coloring problem. In Baum and Landweber [24], pages 87–96.

[6] Erik Winfree, Xiaoping Yang, and Nadrian C. Seeman. Universal computation via self-assembly of DNA: Some theory and experiments. In Baum and Landweber [24], pages 191–213.

[7] Erik Winfree. Whiplash PCR for $O(1)$ computing. In *Proceedings of the* 4th *DIMACS Meeting on DNA Based Computers, held at the University of Pennsylvania, June 16-19, 1998*, pages 175–188.

[8] Martyn Amos, Alan Gibbons, and David Hodgson. Error-resistant implementation of DNA computations. In Baum and Landweber [24], pages 151–161.

[9] Michail G. Lagoudakis and Thomas H. LaBean. 2-D DNA self-assembly for satisfiability. In *Proceedings of the* 5th *DIMACS Meeting on DNA Based Computers, held at the Massachusetts Institute of Technology, June 14-15, 1999*, pages 139–152.

[10] F. Guarnieri, M. Fliss, and Carter Bancroft. Making DNA add. *Science*, 273(5272):220–223, 1996.

[11] Q. Ouyang, P. D. Kaplan, L. Shumao, and A. Libchaber. DNA solution of the maximal clique problem. *Science*, 278:446–449, 1997.

[12] Nobuhiko Morimoto, Masanori Arita, and Akira Suyama. Solid phase DNA solution to the Hamiltonian Path Problem. In Rubin and Wood [25], pages 83–101.

[13] Thomas H. Leete, Joshua P. Klein, and Harvey Rubin. Bit operations using a DNA template. In Rubin and Wood [25], pages 159–171.

[14] Kensaku Sakamoto, Daisuke Kiga, Ken Komiya, Hidetaka Gouzu, Shigeyuki Yokoyama, Shuji Ikeda, Hiroshi Sugiyama, and Masami Hagiya. State transitions by molecules. In *Proceedings of the* 4th *DIMACS Meeting on DNA Based Computers, held at the University of Pennsylvania, June 16-19, 1998*, pages 87–99.

[15] Julia Khodor and David K. Gifford. Design and implementation of computational systems based on programmed mutagenesis. In *Proceedings of the* 4th *DIMACS Meeting on DNA Based Computers, held at the University of Pennsylvania, June 16-19, 1998*, pages 101–107.

[16] Dirk Faulhammer, A. R. Cukras, Richard J. Lipton, and Laura F. Landweber. When the knight falls: On constructing an RNA computer. In *Proceedings of the* 5th *DIMACS Meeting on DNA Based Computers, held at the Massachusetts Institute of Technology, June 14-15, 1999*, pages 1–7.

[17] Junghei Chen, Eugene Antipov, Bertrand Lemieux, Walter Cedeño, and David Harlan Wood. *In vitro* selection for a Max 1s DNA genetic algorithm. In *Proceedings of the* 5th *DIMACS Meeting on DNA Based Computers, held at the Massachusetts Institute of Technology, June 14-15, 1999*, pages 23–37.

[18] H. Yoshida and A. Suyama. Solution to 3-SAT by breadth first search. In *Proceedings of the* 5th *DIMACS Meeting on DNA Based Computers, held at the Massachusetts Institute of Technology, June 14-15, 1999*, pages 9–20.

[19] Masahito Yamamoto, Jin Yamashita, Toshikazu Shiba, Takuo Hirayama, Shigeharu Takiya, Keiji Suzuki, Masanabu Munekata, and Azuma Ohuchi. A study on the hybridization process in DNA computing. In *Proceedings of the* 5th *DIMACS Meeting on DNA Based Computers, held at the Massachusetts Institute of Technology, June 14-15, 1999*, pages 99–108.

[20] Q. Liu, L. Wang, A. G. Frutos, A. E. Condon, R. M. Corn, and L. M. Smith. DNA computing on surfaces. *Nature*, 403:175–179, 2000.

[21] Sam Roweis, Erik Winfree, Richard Burgoyne, Nickolas V. Chelyapov, Myron F. Goodman, Leonard M. Adleman, and Paul W. K. Rothemund. A sticker-based model for DNA computation. *Journal of Computational Biology*, 5(4):615–629, 1998.

[22] Kalim Mir. A restricted genetic alphabet for DNA computing. In Baum and Landweber [24], pages 243–246.

[23] A. R. Cukras, Dirk Faulhammer, Richard J. Lipton, and Laura F. Landweber. Chess games: A model for RNA-based computation. In *Proceedings of the* 4th *DIMACS Meeting on DNA Based Computers, held at the University of Pennsylvania, June 16-19, 1998*, pages 27–37.

[24] Eric B. Baum and Laura F. Landweber, editors. *DNA Based Computers II: DIMACS Workshop, June 10-12, 1996*, volume 44 of *DIMACS: Series in Discrete Mathematics and Theoretical Computer Science.*, Providence, RI, 1998. American Mathematical Society.

[25] Harvey Rubin and David Harlan Wood, editors. *DNA Based Computers III: DIMACS Workshop, June 23-25, 1997*, volume 48 of *DIMACS: Series in Discrete Mathematics and Theoretical Computer Science.*, Providence, RI, 1999. American Mathematical Society.