JOURNAL OF COMPUTATIONAL BIOLOGY
Volume 5, Number 2, 1998
Mary Ann Liebert, Inc.
Pp. 255–267

A Surface-Based Approach to DNA Computation

LLOYD M. SMITH,[1] ROBERT M. CORN,[1] ANNE E. CONDON,[2] MAX G. LAGALLY,[3]
ANTHONY G. FRUTOS,[1] QINGHUA LIU,[1] and ANDREW J. THIEL[1]

ABSTRACT

A scalable approach to DNA-based computations is described. Complex combinatorial mixtures of DNA molecules encoding all possible answers to a computational problem are synthesized and attached to the surface of a solid support. This set of molecules is queried in successive MARK (hybridization) and DESTROY (enzymatic digestion) operations. Determination of the sequence of the DNA molecules remaining on the surface after completion of these operations yields the answer to the computational problem. Experimental demonstrations of aspects of the strategy are presented.

Key words: DNA computation, surface-based, strategy.

INTRODUCTION

THE FIELD OF DNA COMPUTING was initiated in 1994 with a seminal paper by Adleman (1994), in which he proposed that tools of molecular biology could be used to solve computational problems. A modest proof-of-principle was performed by solving a tiny instance of the Hamiltonian Path problem using a test tube-based approach. Although this experimental implementation of DNA computing was an important demonstration, the test tube methodology employed is not well-suited for scale-up to large combinatorial problems, involving necessarily inefficient transfer and handling steps performed on a large scale, with macroscopic volumes and amounts of material.

In this paper we describe an alternative implementation of DNA computing in which complex combinatorial mixtures (the "combinatorial space") of DNA molecules are immobilized on a surface and subsets are tagged and enzymatically modified in repeated cycles of the "DNA computation." When the DNA computation is complete, the sequence of the DNA molecules remaining is determined, yielding the computational result. A schematic depiction of the process is shown in Figure 1.

DNA COMPUTING STRATEGY

1) Solution vs. Surface Chemistry-Advantages of the Solid-Phase Approach.

There are two general formats in which complex combinatorial sets of DNA molecules may be manipulated: (i) in solution (solution-phase format), or (ii) attached to a surface (solid-phase format). It is our belief that the solid-phase format possesses many important advantages over the solution-phase format. These advantages include:

[1]Department of Chemistry, [2]Department of Computer Sciences, and [3]Department of Materials Science and Engineering, University of Wisconsin, Madison, WI 53706.

a) Facilitated sample handling. With the DNA molecules attached to a support, the experimental manipulations are very simple. They are (i) addition of a solution to the support (e.g., for hybridization to or enzymatic modification of surface-bound molecules—see below), and (ii) removal (washing) of a solution from the support. These steps are readily automated, which is essential for eventual practical application of the technology, in contrast to the relatively cumbersome and impractical automation required for manipulations in solution-phase.

b) Decreased losses during sample handling. Inasmuch as the DNA molecules are attached to the surface, losses incurred during the DNA computing process (assuming properly designed surface attachment chemistry) are minimal, in contrast to the inevitable losses accompanying bulk solution transfers. This removes a major source of error and variability in the process.

c) Reduction of interference between oligonucleotides; for example, two complementary sequences, once immobilized upon the surface, will not be able to bind together to form duplexes.

d) Most important of all, solid-phase chemistry permits facile purification of the DNA molecules at every step of the process. When the DNA molecules are attached to a surface, simple washing of the surface with water or buffer removes all species present in solution; left-over enzyme, reaction products and by-products, salts, buffers, and other contaminating species are removed, regenerating a chemically pure set of surface-bound DNA molecules for the next cycle of manipulation. This is absolutely critical to having versatility and control of the molecular biologic reaction steps; in solution-phase it would be difficult or impossible, or at best very slow and inefficient, to attempt to remove salts, enzymes, or other contaminating materials from the DNA molecules. As the conditions required for the various steps to be employed are different (for example, hybridization conditions may differ from exonuclease degradation conditions), it is essential that one be able to readily change these conditions by change of buffers.

The above discussion highlighting the essential need for solid-phase approaches has a generality extending far beyond the present work and is in fact widely accepted in the biotechnology arena: protein sequencing, DNA synthesis, peptide synthesis, even gas and liquid chromatography are all examples of very successful and important automated processes based upon the use of solid phases (Smith, 1988). We believe strongly that DNA computations will not be possible to implement practically except by use of a solid-phase approach, and accordingly the first major tenet of our strategy is to work in a solid-phase format.

Performance of these operations on surfaces, as opposed to in solution, does however introduce new issues of both a fundamental and a practical nature. These issues include a) the negligible diffusion rates for molecules affixed to a support of macroscopic dimensions, affecting hybridization kinetics; b) the high density of negative charge present on a surface covered with DNA, which will tend to repel negatively charged complements in solution, possibly affecting both the kinetics and thermodynamics of hybridization; c) possible steric inaccessibility of support-bound polynucleotides to complementary strands and/or DNA modification enzymes; and d) the requirement for rugged, stable, and well-behaved surfaces and surface attachment chemistry.

2) Nature of the Surface.

Having established the desirability of immobilizing the "combinatorial space" of DNA molecules on a surface, the next issue to be addressed is the type of surface to be employed. Many possible surfaces are available; these include (a) particle supports such as polystyrene or agarose beads or various types of chromatography media, (b) microtiter plates such as 96-well polystyrene or polycarbonate plates widely used in clinical diagnostic assays, and (c) planar supports such as glass microscope slides or silicon wafers. We have chosen the latter based upon the belief that this project demands extremely good control and understanding of the surface chemistry and properties, and that in order for us to develop this control and understanding it is essential to be able to employ a wide variety of analytical methodologies for characterization of the surface properties. This requirement is not met by either particle supports or microtiter plate supports, in marked contrast to planar supports. Particle supports are intrinsically heterogeneous in a chemical sense, both with respect to the dispersion of particle sizes, shapes, and surface properties, as well as with respect to surface functional groups. They are difficult to characterize at an atomic or molecular level. Similar problems apply to microtiter plates which are fabricated from bulk polymer (plastic) materials. In contrast, planar supports can be much more homogeneous (consider for example a wafer of atomically flat silicon) and are amenable to careful characterization by a plethora of analytical techniques such as Fourier transform infrared spectroscopy (FTIR), ellipsometry, surface plasmon resonance (SPR), scanning probe microscopies (STM, AFM, NSOM), and various electron and x-ray diffractive and spectroscopic methods (LEED, XPS, Auger spectroscopy, X-ray diffraction). For these reasons we are presently working with planar glass, silicon, and gold (deposited on glass) surfaces.

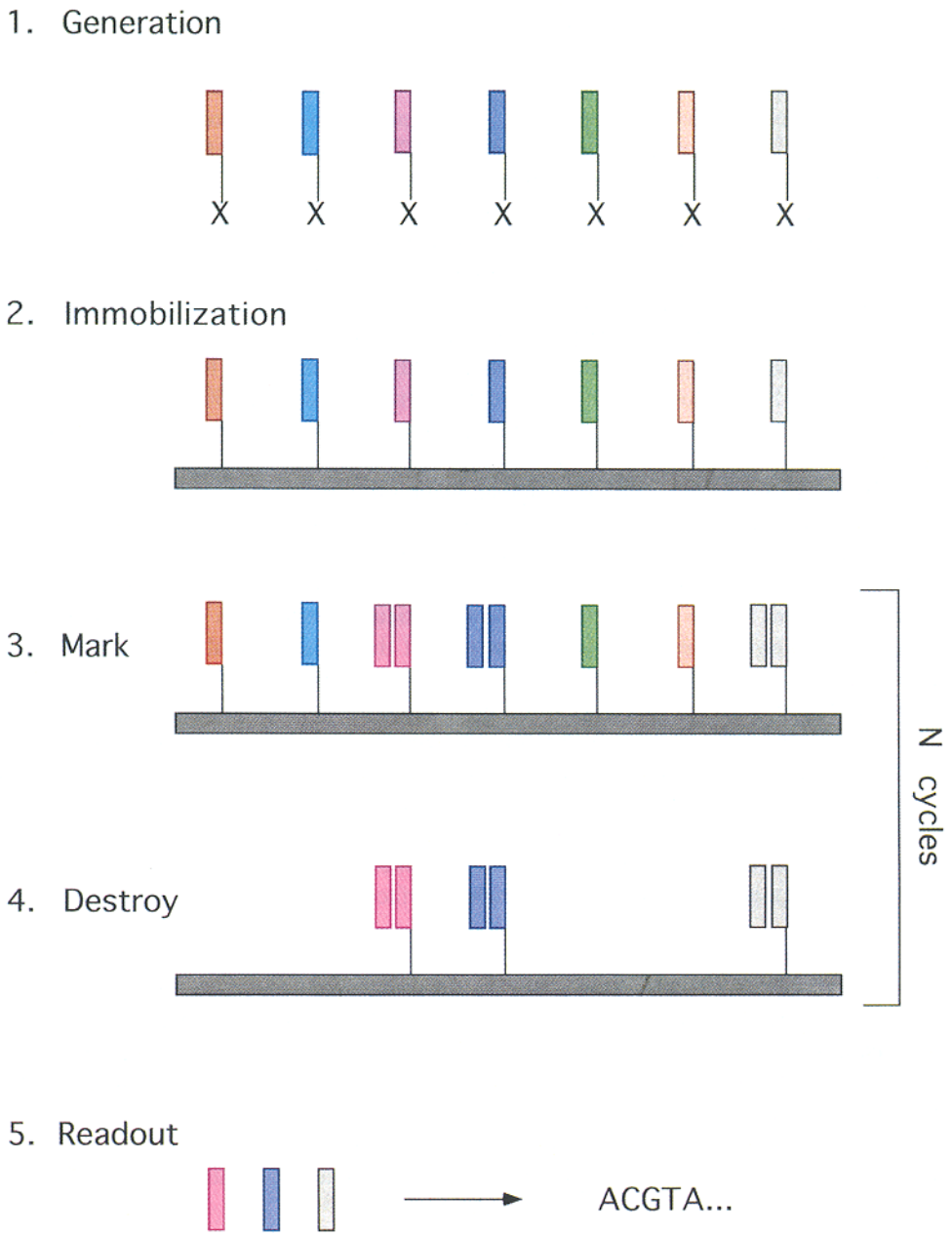

FIG. 1. Overview of the surface-based approach to DNA computations. The different colored bars represent different DNA sequences. A combinatorial set of oligonucleotides is synthesized (Generation) and attached to a support (Immobilization) via a reactive functionality X. In each of N successive cycles of the DNA computation, a solution containing a set of oligonucleotides complementary to a subset of the surface-immobilized mixture is added to the surface and hybridization occurs (MARK); the surface is then washed to remove unbound material and an enzyme (e.g., E. coli exonuclease I) is added which destroys surface-bound oligonucleotides present in an unhybridized (single-stranded) form (DESTROY). After this procedure is repeated N times (using a different set of MARKing oligonucleotides each time) the sequence(s) of the surface-bound oligonucleotides remaining is determined (READOUT), yielding solution(s) to the computational problem.

3) Information Density.

The arguments presented above lead to the conclusion that the combinatorial sets of DNA molecules will be immobilized and manipulated attached to planar supports. An important consequence of this decision is the effect it has upon the available combinatorial complexity, as fewer DNA molecules can be accommodated in a two-dimensional layer than in a three-dimensional volume. In this section the combinatorial space available will be analyzed; it has two major components: (a) the number of different DNA strands ("strings") that can

be attached to the surface (number of strings), and (b) the efficiency with which information can be encoded in the strands (bits per string).

a) Size of the combinatorial space—number of strings. We have considered two different formats for information encoding on the surface: "addressed" mode, and "non-addressed" mode. "Addressed" mode refers to methods of the type pioneered by the company Affymax and later Affymetrix (Fodor *et al.*, 1991; Chee *et al.*, 1996). These workers showed that it is possible to use photolithography in association with solid-phase DNA synthesis chemistry to construct complex arrays of oligonucleotides on glass surfaces. For example, if one makes individual square elements (a sort of DNA pixel) 100 μm by 100 μm on the surface, then a 1 cm^2 "chip" can accommodate 10,000 different oligonucleotides. This methodology is termed "addressed" because each DNA string is synthesized with a known sequence at a known position in the array; i.e., the "address" of that string is known. Although this is an extremely elegant and powerful chemistry with manifold applications, the number of strings available by such methods does not appear to be sufficiently great to have an impact upon the field of computing. The smallest pixels made to date are 50 micron by 50 microns, with array sizes of about a million elements. Even with order-of-magnitude improvements, the array complexity will be well below a billion. Although a billion variables is certainly an impressive number, it is not sufficient to permit a truly significant step forward in computational power.

Because of the above limitation in combinatorial space size using an "addressed" mode, we have chosen the alternative of a "non-addressed" mode. A combinatorial set of DNA molecules is synthesized by standard DNA synthetic methods (more detail on this is provided below) and immobilized upon the support. One does not know where a DNA strand of any particular sequence is on the support (hence "non-addressed"), but one knows that it was formed as part of the combinatorial mixture and accordingly is represented on the support in some number of copies. This changes the spatial scale of information encoding from that determined by the technical limits of photolithography, somewhere on the micron scale, to that of a single DNA molecule, somewhere on the nanometer scale. The advantage of this is a significant increase in the available combinatorial space. The number of strings encoding different solutions on the surface is now of the order of the total surface area divided by the area per DNA strand. A surface area of 100 cm^2 accommodating DNA strands occupying 25 nm^2 can hold over 10^{14} molecules. Once the DNA computing process has been established on relatively ideal planar supports, it may be possible to adapt the process to other support types, providing increased surface area and hence an increased combinatorial space.

Although the advantage of non-addressed mode is an increase in the number of strings which can be represented on the surface, the disadvantage is that one does not know which molecules are left on the surface after performing the various manipulations involved in the DNA computation. One needs eventually to get the answer back out, which means that the sequences of the selected surface-bound DNA molecules must be determined. It may be noted here that the number of molecules attached to the surface is the absolute upper limit for the number of strings that can be employed, only achievable if a single copy of a given solution is enough for the identity of the solution to be determined. It is more likely that many copies of each DNA solution will be needed, decreasing proportionally the available combinatorial space. This issue will be explored in more detail in section 5 below (Readout strategies).

b) Size of the combinatorial space-bits per string. The second major component of the combinatorial space is the efficiency of encoding information within the molecules themselves. To take advantage of the combinatorial complexity available in DNA it is necessary to have a strategy for synthesizing such combinatorial sets of molecules. The chemistry required for generation of these combinatorial mixtures depends strongly upon the manner in which information is encoded within the molecules. In turn, the manner in which information is encoded within the molecules depends strongly upon the specificity and discrimination available in the operations employed for the computing process (e.g., DNA hybridization, enzymatic modification or destruction—see Section 4 below). We have considered several ways to encode information in the DNA molecules, including:

i) The most efficient possible encoding, which makes use of the full base 4 (A, C, G, T) information-carrying capacity of DNA at the single nucleotide level. Although this is the most attractive approach with respect to the high density of information encoding and the ease of synthesis of combinatorial mixtures (see below), the significant variation in thermal stability of different duplexes, as in, for example, the well-known effect of G/C content, make this approach problematic.

ii) Base 2 encoding in which at a given position the sequence is either C or G, or either T or A. Although this has the disadvantage of decreasing the density of stored information and hence necessitates the use of longer oligomers, it has the advantage of permitting the G/C content of the mixtures to be kept constant

by controlling the relative numbers of C or G vs T or A positions. Results obtained with this approach are presented in an adjoining paper (Liu *et al.*, 1998).

iii) Encoding with multiple bases per bit, providing greater discrimination in the hybridization process, albeit at the cost of information encoding density and requiring somewhat more complex generational methodology.

c) Synthesizing combinatorial mixtures. It is straightforward using today's technology to make the needed combinatorial mixtures. Consider the simple problem of representing all possible bit-strings of length n as DNA strands. In the widely available automated solid-phase phosphoramidite DNA synthesis chemistry a desired DNA molecule is built up nucleoside by nucleoside on a support particle in sequential coupling steps (Smith, 1988). For example, a support with the nucleoside "A" attached may have the "A" reacted with a "C" to form a dimer, and the "C" coupled with "G" to form a trimer (still attached to the surface) and so on. After the desired number of coupling steps, the oligomer is cleaved from the surface and is ready for use in experiments. This same chemistry can produce combinatorial sets of molecules by using mixtures of nucleosides at each coupling step. For example, if all four nucleosides are used together in three subsequent coupling steps, 64 different molecules (4^3) are made on the support. These DNA molecules are generally synthesized on a micromole scale for routine laboratory research: as a micromole is 6×10^{17} molecules, the maximum combinatorial complexity available (with an average of one molecule of each type) is 6×10^{17}, which corresponds to a set of all possible 30mers. The variables in this example are coded in base 4 (the four nucleosides A, C, G, T); coded in base 2 (e.g., using oligonucleotides with either C or G, or T or A, at a position), a micromole of DNA corresponds to a combinatorial set of all possible 59mers. Thus a set of molecules required for a 59-binary-variable SAT problem can be synthesized in an afternoon, on commercially available equipment, for a cost of about $100. If the scale of the synthesis is increased by a thousand-fold to yield a millimole of product (a standard procedure in the emerging DNA-based biopharmaceutical industry), the number of binary variables possible increases to 69.

A noteworthy limitation of this chemistry is the decreasing purity of the DNA molecules as one goes to longer sequences: for example, with an efficiency of the chemistry at each step of 99.5% (a common result with standard methods), synthesis of a 100mer will yield only $0.995^{100} = 0.60$, or 60% yield of the correct sequence, the balance being impurities and side products. For this reason it is advantageous to use as high a density as possible of information encoding in the DNA molecules, permitting the oligonucleotides to be as short, and hence as pure, as possible.

In the case of multiple-base encoding direct combinatorial synthesis as described above is not feasible. However, the "word" strategy described in section d) below does lend itself to a generational strategy based upon standard DNA synthetic methods. This is described in more detail in section d) below.

d) A DNA "word" strategy. One problem that arises in considering the above strategies for DNA computing is their extensibility to increasingly complex mixtures. More complex mixtures require longer oligonucleotides. As will be discussed further below, selection of particular sequences within the combinatorial mixture involves hybridization, and if information is encoded at the single base level, the hybridization must be able to discriminate a perfectly matched sequence from a single-base mismatch. This becomes increasingly difficult as the duplex becomes longer and longer, as the relative contribution to duplex stability of the single base in question becomes less and less significant. The same issue is operative even if information is encoded in multiple bases, rather than just one—the energetic destabilization afforded by the mismatch becomes less and less significant the longer the oligomer.

We have addressed this issue by means of a DNA "word" strategy for encoding information. Each "word" contains a number of "variable" bases for encoding information, and a number of "fixed" bases of defined sequence that permits complementary oligonucleotides to be targeted specifically to that "word." Multiple words may be concatenated in the combinatorial oligonucleotide mixtures and hybridized to independently with complementary word-specific oligonucleotides, permitting scaling of the approach by use of increasing numbers of words.

To make this more concrete, consider the following example of a word design. Each DNA word consists of sixteen nucleotides (nt); 8 of the nucleotides are variable (i.e., either G/C or A/T; this is a binary variable base encoding scheme), and the other 8 nucleotides are used to label the word. A complete combinatorial set of all possible words with a given label consists of $2^8 = 256$ molecules. The label sequence is invariant for all members of a particular word set so that it can be "queried" by hybridization to complementary 16mers in the presence of other word sets. The words may be "linked" together when synthesized, and attached to the surface as short "sentences" of 2, 3, 4, or more consecutive words. The corresponding combinatorial space size is 2^{16} (= 65,536, for two words), 2^{24} ($= 1.7 \times 10^7$, for three words), or 2^{32} ($= 4.3 \times 10^9$, for four words)

molecules respectively. Further increases in combinatorial space are readily obtained by adding additional words. This approach provides us with several important advantages, as follows:

i) Once we have designed the 8 nt combinatorial set of oligonucleotides for a single word, that same set, with different labels for new words, may also be employed for the other words. The different words of the "sentence" can be queried by hybridization either at the same time, or serially, as desired.

ii) Secondary structure and hybridization discrimination issues associated with longer oligonucleotides will be reduced. Longer oligonucleotides have a much greater propensity to form secondary structures than do shorter ones; without a word strategy we would be likely to run into ever-increasing problems with both secondary structure and single base hybridization discrimination as we go to longer combinatorial strings.

iii) The use of these relatively short words will allow us to read out the answers a word at a time by hybridization to a combinatorial *addressed* array of 256 oligonucleotides (complementary to our 256 member combinatorial non-addressed set). This provides an interesting and possibly important alternative to conventional sequencing for readout (see Section 5 below). This word strategy provides a path for scaling up the complexity of the DNA computations in a much simpler and more straightforward fashion than would otherwise be possible.

As mentioned in section c) above, the word strategy also provides a path for the generation of the combinatorial mixtures needed in a multiple-base encoding strategy. This may be done as follows. Individual short strings corresponding to various possible words are synthesized separately on the standard solid-phase DNA synthesis support particles. The supports to which they are attached are combined, and the mixture is redistributed into new columns for continued solid-phase synthesis of the next word. This process is continued for each word. For example, if a given word has sixteen different states, then sixteen parallel oligonucleotide syntheses would be performed, one for each different state of the word. The supports containing these oligonucleotides would be combined into a mixture, divided into sixteen portions, and synthesis of the next sixteen individual words would be performed using this material as the support. These supports when combined would have a complexity of $16^2 = 256$; ten such cycles would yield a complexity of $16^{10} = 10^{12}$. Thus quite a respectable complexity may be obtained using fairly straightforward and well-established DNA synthesis chemistry.

4) Operations on Surfaces.

The heart of the DNA computing process is the selective recognition and enzymatic manipulation of DNA molecules. The repertoire of the molecular biologist in this regard is powerful: standard procedures in molecular biology include hybridization, sequence-specific cleavage using restriction endonucleases, selective destruction of single-stranded or double-stranded DNA with exonucleases, ligation of DNA strands to one another, sequence-specific methylation, phosphorylation of the 5′ terminus, and extension of the 3′ terminus. This large repertoire of possibilities makes possible a similarly large repertoire of DNA operations. In this section a simple version of surface-based DNA computing is illustrated with three basic operations: MARK, UNMARK, and DESTROY.

In the MARK operation a desired combinatorial mixture of DNA strands is added to the surface: those strands that find a complement on the surface will bind to form a duplex: thus MARKED strands will be duplex, and UNMARKED strands will be single-stranded. For example, suppose the query was "NOT A at nucleotide 9"; the complementary mixture would contain every possibility at each position except 9, whereas at 9 only one of the two possibilities would be present. All of the strands except for one would be MARKED. When there are multiple words in the DNA string, the MARK operation will form a duplex only in part of the string. This duplex region can be recognized as is in subsequent enzymatic steps (e.g., restriction enzyme cleavage). Alternatively it may be extended with a polymerase to make a longer duplex region (when multiple words are employed this step is necessary for use of the exonuclease-based DESTROY operation below). The feasibility of encoding information at the single nucleotide level for performing the MARK operation is studied in the adjoining paper (Liu *et al.*, 1998).

The DESTROY command in this example consists of adding an exonuclease specific for single-stranded DNA. Every unmarked strand is destroyed, leaving on the surface only the MARKED DNA molecules.

The UNMARK command consists of subjecting the surface to conditions under which hybrids dissociate into single strands. This may be done by a combination of increased temperature and addition of denaturants such as urea. Subsequent washing with a suitable buffer removes the free strands in solution and regenerates the DNA-modified surface.

After each cycle of MARK, DESTROY, and UNMARK, fewer molecules remain on the surface. Repeated queries (e.g., one hundred cycles of MARK, UNMARK, and DESTROY commands) constitute the DNA computation process, permitting subsets of the initial combinatorial space to be eliminated, and leaving the desired solutions to the problem of interest.

This simple example serves to illustrate the basic principle of DNA computations on surfaces. However it may be noted that the computational power of the process is greatly increased if we are able to modify the information content in more subtle ways than complete destruction. An important aspect of the development of this technology will be the implementation of additional DNA recognition/modification operations. One of particular interest is an APPEND operation, which could be implemented using the enzyme ligase, which can covalently and specifically join two DNA strands. Together, the MARK, UNMARK, DESTROY and APPEND operations form a "computationally complete" set in that exactly those strands (representing binary strings) on the surface that satisfy a given Boolean circuit can be identified (i.e., MARKED) with a number of operations that is proportional to the size of the circuit (Cai *et al.*, 1997).

It should also be noted that there are important differences in the operations available in the surface model and solution model. For example, when using planar supports the SEPARATE and MERGE commands possible in solution-phase (Adleman, 1994) are not feasible. These latter commands permit more complex operations: however, given the present practical limitations of the solution-phase model, we are willing to accept this limitation of the solid-phase model. The use of particle supports rather than planar supports may permit these commands to be enabled in the future.

We will use the famous Satisfiability (SAT) problem (Garey and Johnson, 1979) to illustrate how the MARK, UNMARK and DESTROY operations can be used to "compute." First we describe the SAT problem using a simple example:

(x or y or not {z}) and (w or not {z}) and (v) and (not {w} or not {y}).

This example contains four "clauses," delimited by parentheses, over the five "variables" v, w, x, y, and z. The first clause is "satisfied" if either x or y is true or if z is false, the second clause is satisfied if either w is true or z is false, and so on. More generally, a SAT formula consists of clauses over a large set of variables, and the SAT problem is to determine if there is an assignment of true/false values to the variables that satisfy all clauses simultaneously. In our example, there are $2^5 = 32$ possible ways to assign truth values to the 5 variables. It is not hard to check that of these 32 possibilities, only 7 actually satisfy all clauses. More generally a SAT formula with N variables has 2^N possible truth assignments, and examining an exponential number of these possibilities becomes infeasible on any existing computing device even for values of N less than 100. Unfortunately, the SAT problem is known to be "NP-hard" (Garey and Johnson, 1979), which indicates that no algorithm, no matter how sophisticated, can overcome this exponential barrier. Now, suppose that each strand on a surface represents a truth assignment of the variables of a SAT formula. (A binary string represents a truth assignment in a natural way, with 0 representing "false" and 1 representing "true".) The following algorithm destroys all strands that do not satisfy a given SAT formula, leaving on the surface exactly those strands that satisfy the formula.

```
repeat for each clause C:
   for each unnegated variable v in C do
      MARK all strands in which v is set to "true"
   for each negated variable v in C do
      MARK all strands in which v is set to "false"
   {comment: now, all strands that satisfy the clause C are MARKED}
   DESTROY
   UNMARK
endrepeat
```

On our example SAT formula, the outer repeat loop would be executed four times. On the first execution, all strands which do not satisfy the first clause are destroyed, namely those 4 strands in which both x and y are set to "false" and z is set to true. On the second execution, a further 6 strands are destroyed, namely those left on the surface in which w is set to false and z is set to true. A further 11 and 4 strands are destroyed on the third and fourth executions, respectively, leaving the 7 satisfying assignments on the surface.

Because the algorithm operates on all strands on the surface in parallel, the amount of "computation" done in this way can be measured as roughly the number of DNA operations times the number of strands on the surface. Of course, like all known algorithms for SAT, our algorithm does not overcome the "exponential barrier," since the number of strands needed on the surface for an instance with N variables is 2^N. It is intended simply to illustrate how parallel computations can be done using DNA on surfaces.

5) Readout Strategies.

The final aspect of this strategy to be discussed is that of Readout. The end result of a DNA computation in this model is a surface on which DNA molecules exist, whose sequence encodes the solution to a combinatorial problem of interest. It is thus necessary to determine the sequence(s) of these surface-bound DNA molecules in order to ascertain the solution to the computational problem in question. Two approaches to this problem are a) conventional electrophoresis-based DNA sequencing and b) hybridization to word-specific addressed arrays. It may be noted that most problems will have multiple solutions rather than a single unique solution. This complicates readout; for example, in the case of multiple solutions conventional sequencing of the molecules remaining on the surface will yield ambiguous results due to the presence of multiple bases at various positions. For simplicity we will assume here that it is sufficient to determine a single solution to the problem.

It will generally be necessary to amplify the number of copies of the solution in order to determine their identities. This may be done either by conventional cloning, or by using the Polymerase Chain Reaction (PCR, Mullis and Faloona, 1987). In the latter case appropriate PCR-primer sequences are designed into the strings to permit their eventual amplification. An advantage of cloning is that by its nature it isolates individual solutions to the problem; however substantial additional time and effort are required for cloning compared to PCR amplification.

a) Readout by fluorescence-based automated DNA sequencing. If there is a unique solution, or the solutions have been cloned, direct sequencing will provide the solution's identity. If there are multiple solutions and the material is PCR-amplified, sequencing will reveal a number of possible solutions which may then be checked by conventional computing methods. A suitable primer sequence, which can be the same as one of those employed for PCR amplification, is included in the string design to permit eventual enzymatic sequence analysis.

b) Readout by hybridization to addressed arrays. An alternative is to employ addressed hybridization arrays for readout of the computational result on a word-by-word basis. Such arrays can be fabricated manually, robotically, or by photolithography (Fodor *et al.*, 1991; Chee *et al.*, 1996). In one interesting approach to this, the complementary strands on the surface are melted off, PCR-amplified with fluorescent tags, and hybridized to the addressed array corresponding to word 1. A fluorescently tagged pixel indicates that the corresponding word value is present. All strings on the original unaddressed surface not having this sequence are then MARKED and DESTROYED, leaving a reduced combinatorial space with a common word 1. This is repeated for word 2, word 3, and so on, finally yielding a particular solution to the problem.

6) Error Control.

The intrinsic complexity and variability of the chemical processes to be utilized in DNA computations ensure that perfection will not be attained in the individual operations. Rather, errors will occur, and accordingly the management of error is an essential aspect of the system design and development. The development of error models that occur in DNA computations of this sort, and of algorithmic methods to control these errors is essential. As early prototypes of DNA computers are developed along the lines outlined above the analysis of the errors and imperfections of the process will be critical to devising appropriate solutions.

By way of illustration it is useful to consider the simple model outlined above, in which DNA computations are performed by means of MARK, UNMARK, and DESTROY operations. The most serious problem caused by errors in this model is due to a combination of the mark and destroy (unmarked strands) operations. Errors in the mark operation, due for example to insufficient specificity in the DNA hybridization process, lead to "false positives" (strands which should not be marked will be marked) and "false negatives" (strands which should be marked are not marked; see the adjoining paper (Liu *et al.*, 1998) for experimental results demonstrating these effects). The false negatives will be lost irrevocably upon application of the destroy operation. This may be circumvented to some degree by adding redundancy to the combinatorial space. That is, multiple copies of each strand need to be present. Assuming that the false negatives are a random subset of the combinatorial space (an assumption that can be tested experimentally), the probability that all copies of one strand are false negatives decreases as the redundancy increases. It may be noted that this solution is not without cost—since space on a surface is limited, increasing redundancy means that fewer distinct strands can be put on the surface, and this limits the size of problems that can be solved.

EXAMPLES OF OPERATIONS

The key to successful development of the above approach to DNA computing is effective implementation of the operations on surfaces. Our experimental development of the operations is facilitated by use of an addressed format (surface-bound oligonucleotide arrays) rather than an unaddressed format, as with oligonucleotide arrays the efficiency and specificity of hybridization and enzymatic modification procedures are readily and

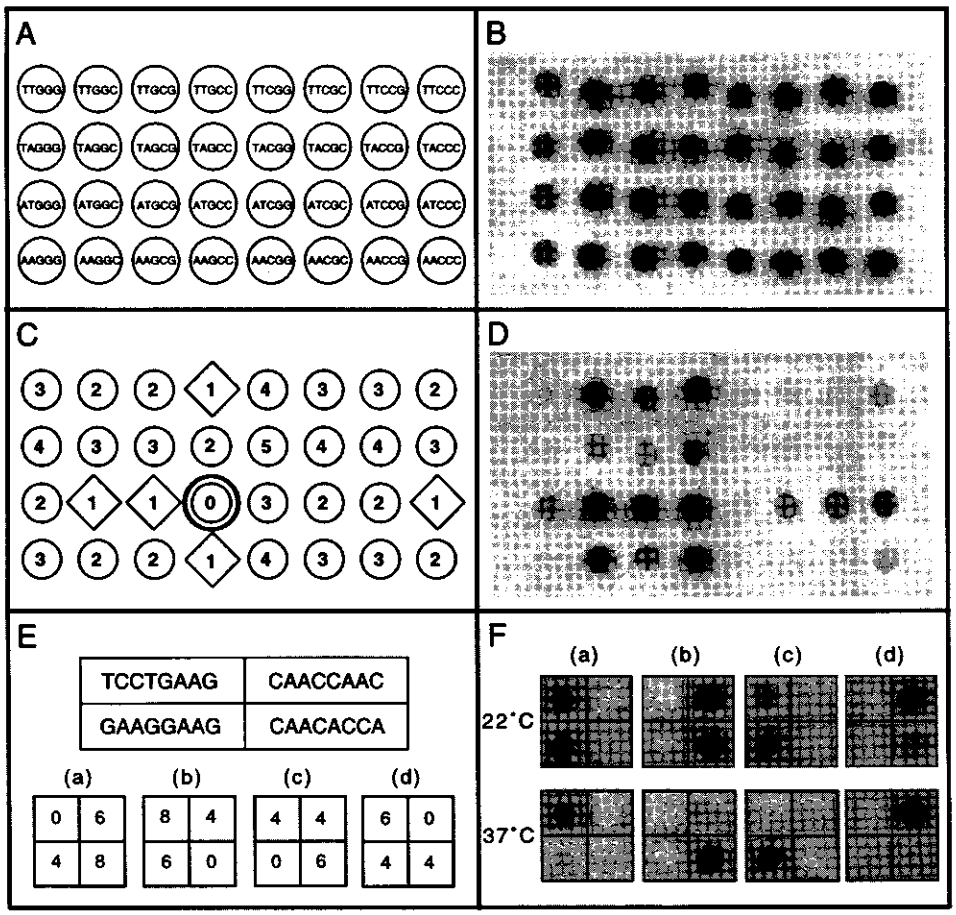

FIG. 2. Results of the MARK (hybridize) operation performed on arrays of surface-immobilized oligonucleotides. All results shown in this and subsequent figures were produced by fluorescence scanning with a Molecular Dynamics FluorImager 575. Panels A–D show the results of the MARK operation when information is encoded at the single-base level, and panels E and F show the results when information is encoded with 4 bases. Panel A shows the sequences of the bases at the five variable positions, for each of the 32 (2^5) members of the array. Panel B shows the results of hybridizing a fluorescently tagged mixture of all 32 complementary oligonucleotides (15 nt in length) to the array. Panel C shows the mismatches obtained when the same array is hybridized to only a single fluorescently tagged 15mer oligonucleotide, perfectly complementary to the sequence at column 4 row 3 of the array, and Panel D shows the results obtained in that experiment. The single-base mismatches are indicated in Panel C by diamonds. Panel E (upper rectangle) shows the sequences of the bases at the eight variable positions employed with the 4-base encoding strategy for the four oligonucleotides employed in these experiments. These four oligonucleotides are immobilized on the surface in the 2 × 2 array format shown. Panel E columns (a)–(d) shows the mismatches present in four separate hybridization experiments with the perfect 16mer complements of one of the four array members, and Panel F shows the corresponding hybridization results obtained at 22°C and 37°C.

directly evaluated by quantitative fluorescence imaging (Guo *et al.*, 1994). Figure 2 shows results obtained in the MARK (hybridization) operation using either a single base information encoding strategy (panels A–D) or a 4 base information encoding strategy (panels E–F). In the single-base encoding approach (Liu *et al.*, 1998), a set of $2^5 = 32$ individual oligonucleotides was synthesized and arrayed on a glass support. These oligonucleotides had the sequence

$$5'NH_2-(T)_{15}-AGGAWWSSSGGTTAT\,3'$$

where W represents T or A and S represents C or G. Thus the variable positions W and S encode information in a binary format while maintaining a constant GC content within the hybridization sequence. The sequences of the five variable bases for each member of the array are shown in panel A.

Panel B of Figure 2 shows the fluorescence intensity pattern obtained when the array is hybridized with a combinatorial mixture of all 32 complementary oligonucleotides fluorescently tagged at their 5′ termini. All 32 members of the array bind a fluorescent complement under the conditions employed. Interestingly, the

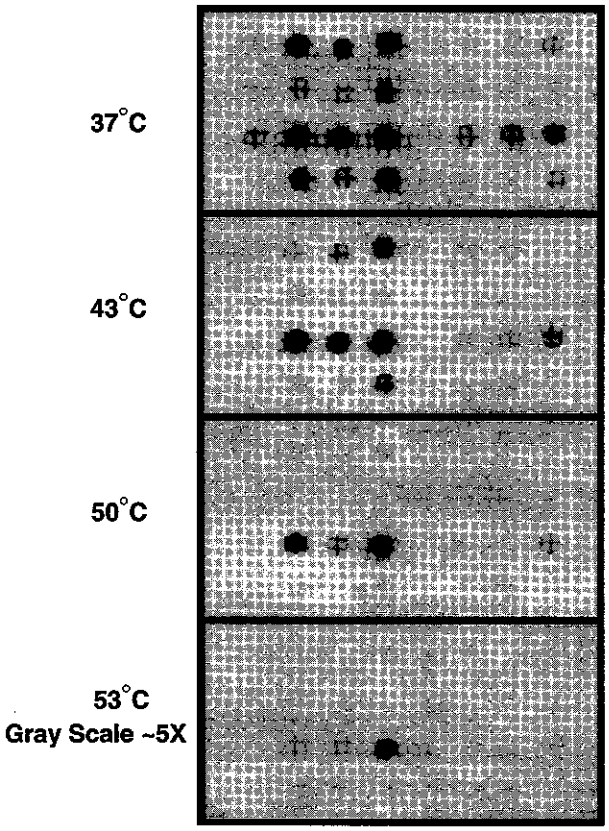

FIG. 3. Results obtained in the MARK operation using single-base encoding under conditions of increasing stringency (temperature). The data in the top panel are identical to the data in Figure 2 Panel D. Each of the four panels shows the fluorescence remaining on the surface after washing at the indicated temperatures of 37°C, 43°C, 50°C, and 53°C, respectively. The grey scale of the bottom panel (53°C) has been adjusted to be approximately 5X darker than the other panels to compensate for the decreased fluorescence remaining after the stringent washing needed to obtain apparent single-base specificity.

leftmost column in panel B shows markedly reduced fluorescence intensity compared to other members of the set. Examination of the sequences involved showed that the three adjacent Gs present in the members of this column can interact with the $(T)_{15}$ spacer by means of relatively stable G-T mismatches (Aboul-ela *et al.*, 1985) to form a hairpin structure within these surface-bound oligonucleotides. Support for this hypothesis was obtained in digestion experiments using E. coli Exonuclease I, a single-strand-specific exonuclease, which did not digest the members of column 1 effectively compared to other members of the array (data not shown). This putative hairpin structure presumably renders these oligonucleotides relatively inaccessible for hybridization to their solution complements, accounting for the reduced fluorescence intensity observed. This interesting result underscores the importance of careful design and evaluation of the combinatorial mixtures to avoid unwanted effects in the hybridization and enzymatic manipulation procedures (Liu *et al.*, 1998).

Panels C and D of Figure 2 show the results obtained when the same array is hybridized to a single fluorescently tagged oligonucleotide complement. Panel C shows the number of mismatched bases for each member of the array, which ranges from zero (perfect match) for the oligonucleotide in column 4 row 3, to all five bases for the oligonucleotide in column 5 row 2. These hybridization results were obtained under relatively low stringency conditions, as is evident from the substantial degree of hybridization observed to mismatched members of the array. The hybridization signal is generally most intense for the perfectly matched and single-base mismatched members of the array as one might expect, although interestingly the single-base mismatch oligonucleotide of column 8 row 3 shows considerably less intensity than does the two base mismatched oligonucleotide of column 2 row 1. A thorough analysis of these results and comparison with thermodynamic predictions are presented in the adjoining paper (Liu *et al.*, 1998). In Figure 3, the results of experiments in which the stringency of hybridization is gradually increased to yield single-base hybridization specificity are presented. Although these results show that under appropriate conditions single-base specificity may be obtained, quantitative analysis of the results shows clearly that this degree of hybridization specificity is obtained at substantial cost—the fluorescence intensity obtained from the perfectly matched oligonucleotide under the 53°C high stringency conditions is approximately 25% of that obtained under the low stringency 37°C conditions. This low efficiency of hybridization under the conditions necessary for single-base specificity demonstrates compellingly that this approach to single-base encoding of information for DNA computing is not viable, at least not in the present form.

To address this problem we have investigated alternative base-encoding schemes providing higher hybridization specificity. Panels E and F of Figure 2 show results obtained using 4-base encoding of information rather than single-base encoding. In these experiments, which were performed on gold surfaces (in all these experiments similar results are obtained with glass, silicon, and gold surfaces—details of the surface attachment chemistries employed were reported elsewhere, Frutos *et al.*, 1997), 16 base hybridization sequences were employed, of sequence

$$5'\,\mathrm{HS}-(\mathrm{T})_{15}-\mathrm{GCTTvvvvvvvvTTCG}\,3'$$

where "v" represents a variable (information-encoding) base position. The maximum combinatorial space encoded in this v_8 region is $4^8 = 65{,}536$. The stipulation that the GC content of all members be maintained at 50% reduces this number to 17,920. The further requirement that all members of the set differ by at least 4 mismatches reduces the possibilities to 108 (Frutos *et al.*, 1997) (the combinatorial complexity which may be achieved with this set of mismatched oligonucleotides is 108^n where n = the number of words employed; for example, with 5 words the complexity is $108^5 = 1.5 \times 10^{10}$). Four members of this set with the variable sequences shown in Figure 2E were selected for evaluation in surface hybridization experiments. Figure 2F shows the results of hybridization experiments using each of the four perfectly matched complements (columns (a)–(d)) and Figure 2E shows in each case the number of mismatched bases present.

The results in Figure 2F shows that in each case complete discrimination between the matched and mismatched sequences may be obtained. Quantitation of the fluorescence signal obtained for the perfectly matched duplexes under the conditions employed for high specificity (37°C wash) shows it to be reduced by less than 5% from the signal intensity obtained under low stringency conditions, demonstrating that this four-base encoding scheme provides both the high efficiency and specificity of hybridization needed for an effective MARK operation.

Figure 4 shows an example of results obtained in three successive cycles of the MARK and DESTROY operations, applied to an array of four 16mer oligonucleotides immobilized upon a surface. In this version

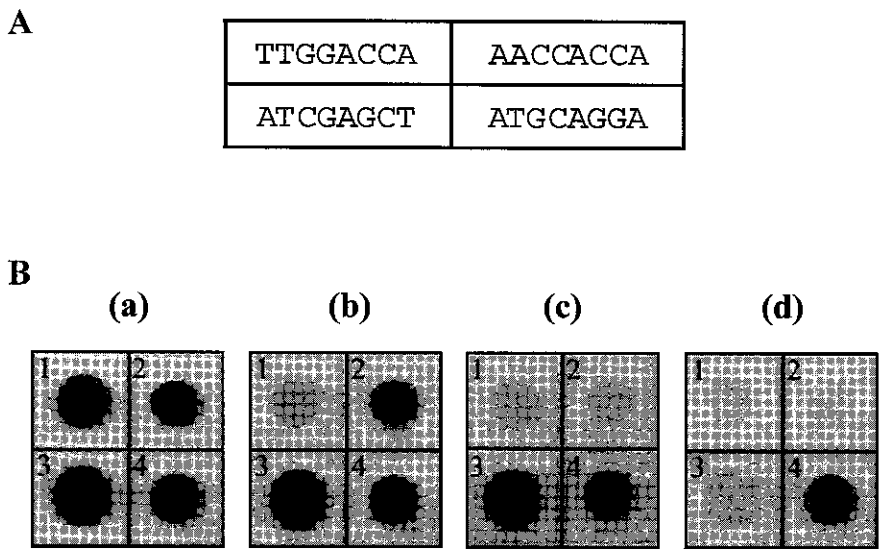

FIG. 4. Three successive DNA computation cycles performed upon an array of four 16mer oligonucleotides. The 16mers have the sequence 5′ GCTTvvvvvvvvTTCG 3′, where the four bases at the 5′ and 3′ termini are "word labels" that are invariant in sequence in this experiment, and the 8 internal bases used to encode information possess four-base mismatches (sequences shown in Panel A). Panel B columns (a)–(d) show the experimental results obtained. (a) The array was visualized by hybridization with a mixture of all four fluorescently tagged perfectly matched complements and subsequent fluorescence imaging; this image shows that all four oligonucleotides are present on the surface and available for hybridization in approximately equal amounts; (b) The same array was washed to remove hybridized complements, MARKED by hybridization with the perfectly matched complements to the oligonucleotides in spots 2–4, and spot 1 (which was not MARKED) was selectively DESTROYED by addition of E. coli exonuclease I. The surface was washed, visualized again by hybridization with the mixture of all four fluorescently tagged perfectly matched complements, and imaged to yield the results shown; this image shows that the MARK and DESTROY operations employed together were able to selectively destroy the desired target sequence; (c) and (d) show the same process repeated, but using in the MARK operation only oligomers complementary to spots 3 and 4 in (c), and only complementary to spot 4 in (d). The results show that the cycles of MARK and DESTROY can be employed repeatedly to successively eliminate subsets of the surface-immobilized oligonucleotides.

of the DESTROY operation, single-strand-specific E. coli Exonuclease I was utilized to selectively destroy UNMARKED surface-bound oligonucleotides. In Panel B, (a) shows that initially all four oligonucleotides are present on the surface and available for hybridization, and (b)–(d) show the results of three successive cycles of MARK and DESTROY targeted successively at the oligonucleotides in spots 1, 2, and 3. The selective retention of the oligonucleotide in spot 4 through this DNA computing procedure demonstrates on a small scale the feasibility of the surface-based approach to DNA computations. Further work will focus upon increasing the complexity of the mixtures, performing experiments in an unaddressed mode rather than an addressed mode, further improving the efficiency of the operations, and developing further operations.

SUMMARY

In this paper we have outlined an approach to performing DNA computations on surfaces, designed to permit many successive cycles of computing operations to be accomplished relatively easily. Automation of the procedure will be relatively straightforward once the necessary chemistry has been developed. However, implementation of this strategy presents significant chemical challenges. The surface chemistry for attachment of the combinatorial DNA mixtures must be extremely stable and not compromise the accessibility of the surface-bound molecules to either a complementary strand for duplex formation, nor to enzymatic modification. The specificity of the computing operations must be exceedingly high, as to be useful for computing, the combinatorial mixtures to be queried must have complexity far beyond even that of the three billion base human genome. Hybridizing complex DNA mixtures in solution, to complex DNA mixtures immobilized on supports, means that the concentration of any given strand may be extremely low, necessitating very long hybridization times. These and other technical issues are non-trivial in nature and substantial work may be required to accomplish even fairly modest proof-of-principle demonstrations of the approach. Proof-of-principle demonstrations of the MARK and DESTROY operations are presented here which illustrate the general feasibility of the approach.

Clearly a great deal of work will be needed for this or other DNA computing approaches to be of practical consequence. In the meantime, however, these ideas offer a tantalizing new model for how to think about the computing process itself. Biology provides countless examples (consider an enzyme, a bacterium, a yeast cell, a neuron, a flatworm, an endocrine gland, a brain, a human) of tremendously powerful "computing devices" which operate in ways completely different than our present concept of "conventional" computing. Such cells, organs, and organisms process astoundingly complex inputs and provide astoundingly complex outputs rapidly and efficiently, in manners into which we are only beginning to gain a small amount of insight. Whether DNA computing becomes a practical way of solving computational problems remains to be seen. What is already clear is that it is stimulating new model for how to think about other ways of computing.

REFERENCES

Aboul-ela, F., Koh, D., Tinoco, I., Jr., and Martin, F.H. 1985. Base-base mismatches. Thermodynamics of double helix formation for $dCA_3XA_3G + dCT_3YT_3G$ (X, Y = A, C, G, T). *Nucleic Acids Res.* 13, 4811–4824.

Adleman, L.M. 1994. Molecular computation of solutions to combinatorial problems. *Science* 266, 1021–1024.

Cai, W., Condon, A.E., Corn, R.M., Glaser, E., Fei, Z., Frutos, A.G., Guo, Z., Lagally, M.G., Liu, Q., Smith, L.M., and Thiel, A.J. 1997. The power of surface-based DNA computation. *Proceeding of the First Annual International Conference on Computational Molecular Biology (Recomb 97)*, ACM, 67–74.

Chee, M., Yang, R., Hubbell, E., Berno, A., Huang, X.C., Stern, D., Winkler, J., Lockhart, D.J., Morris, M.S., and Fodor, S.P.A. 1996. Accessing genetic information with high-density DNA arrays. *Science* 274, 610–614.

Fodor, S.P.A., Read, J.L., Pirrung, M.C., Stryer, L., Lu, A.T., and Solas, D. 1991. Light-directed, spatially addressable parallel chemical synthesis. *Science* 251, 767–773.

Frutos, A.G., Liu, Q., Thiel, A.J., Sanner, A.M.W., Condon, A.E., Smith, L.M., and Corn, R.M. 1997. Demonstration of a word design strategy for DNA computing on surfaces. *Nucleic Acids Res.* 25, 4748–4757.

Garey, M.R., and Johnson, D.S. 1979. *Computers and Intractability: a Guide to the Theory of NP-Completeness.* Freeman Press, N.Y.

Guo, Z., Guilfoyle, R.A., Thiel, A.J., Wang, R., and Smith, L.M. 1994. Direct fluorescence analysis of genetic polymorphisms by hybridization with oligonucleotide arrays on glass supports. *Nucleic Acids Res.* 22, 5456–5465.

Liu, Q., Frutos, A.G., Thiel, A.J., Corn, R.M., and Smith, L.M., 1998. DNA computing on surfaces: Encoding information at the single base level. *J. Comput. Biol.* 5/2.

Mullis, K.B., and Faloona, F.A. 1987. Specific synthesis of DNA *in vitro* via a polymerase-catalyzed chain reaction. *Methods in Enzymol.* 155, 335–350.

Smith, L.M. 1988. Automated synthesis and sequence analysis of biological macromolecules. *Anal. Chem.* 60, 381A–390A.

Address reprint requests to:
Lloyd M. Smith
Department of Chemistry
University of Wisconsin
Madison, WI 53706

Received for publication November 19, 1997; accepted as revised January 21, 1998.