

A DNA Based Implementation of an Evolutionary Search for Good Encodings for DNA Computation

R. Deaton, R. C. Murphy, J. A. Rose, M. Garzon, D. R. Franceschetti, and S. E. Stevens, Jr.*

September 17, 1998

Abstract

Computation based on manipulation of DNA molecules has the potential to solve problems with massive parallelism. DNA computation, however, is implemented with chemical reactions between the nucleotide bases, and therefore, the results can be error-prone. Application of DNA based computation to traditional computing paradigms requires error-free computation, which the DNA chemistry is unable to support. Careful encoding of the nucleotide sequences can alleviate the production of errors, but these good encodings are difficult to find. In this paper, an algorithm for evolutionary computation with DNA is sketched. Evolutionary computation does not require error-free DNA chemistry, and in fact, takes advantage of errors to produce change and variation in the population. An application of the DNA based evolution program to a search for good DNA encodings is sketched.

Introduction

Computation with DNA molecules possesses an inherent interest for researchers in computers and biology. Because of the huge numbers of DNA molecules in a typical test tube, any method of computation based on DNA would seem to have potential massive paral-

lelism, capacity, and power. This potential, however, is limited by the constraints imposed by the DNA chemistry[1].

Adleman[2] introduced a way to solve combinatorial optimization problems with DNA that assumed error-free computation. Because of the chemical nature of the reaction upon which the technique is based, however, error-free results are difficult to guarantee. As implemented by Adleman[2], the fundamental reaction in DNA based computation is hydrogen bonding between Watson-Crick complement base pairs, $\overline{A} \equiv T$ and $\overline{G} \equiv C$ [3]. The chemical process in which two single strands of DNA (oligonucleotides) are hydrogen bonded together is called hybridization. In Adleman's original work[2], a Hamiltonian path through a graph was formed through successive hybridizations of oligonucleotides (oligos) which represented vertices and edges in the graph. Subsequent proposals have continued to use the mechanism of hybridization to do computation[4, 5, 6, 7]. The exact products of hybridizations among a set of oligonucleotides depend upon the reaction conditions. Oligonucleotides that are not perfect complements of each other can bind[8], producing an error. The closer the oligos are to being perfect complements, the more likely is their hybridization. In previous work[1], a set of oligo encodings was characterized with the Hamming distance. In order to prevent the possibility of errors, it was proposed that the oligos in a computation be some distance apart, which depends on reaction conditions. This leads immediately to the Hamming bound as an upper limit on the size of a DNA based computation[1].

Algorithms that use evolution, or survival of the fittest, as inspiration include genetic algorithms, genetic programs, and evolutionary programs[9]. An initial population is chosen, randomly. Based upon

*R. Deaton and J. A. Rose are with The Department of Electrical Engineering, The University of Memphis, Memphis, TN, 38152, r-deaton@memphis.edu, R. C. Murphy and S. E. Stevens, Jr. are with The Department of Microbiology and Molecular Cell Sciences, The University of Memphis, Memphis, TN, 38152, M. Garzon is with The Department of Computer Science, The University of Memphis, Memphis, TN, 38152, garzonm@cc.memphis.edu, and D. R. Franceschetti is with The Department of Physics, The University of Memphis, Memphis, TN, 38152

their fitness, or how well they optimize or satisfy an external constraint, individuals are selected from a population. Fit parents are selected at random, and children formed from them through crossover. Small changes in individuals in the population are made by random mutation. The basic evolution program (EP) algorithm is shown in Figure 1. Through successive generations, the fitness of individuals is improved, and reasonable solutions to many difficult problems are obtained.

A good set of oligos to use as DNA encodings in a computation do not participate in unproductive hybridizations, such as mismatched hybridizations, hairpins, or shifted hybridizations (Figure 2). Finding such a set of oligos, however, is a difficult problem. The search for a set of oligos that would be good encodings for a DNA computation involves the comparison of many DNA strings. Since there are no analytical methods to calculate the encodings, evolutionary techniques were used to search for a set of good encodings[10], and good results were obtained[1]. Even using a genetic search, the task of computing a set of good DNA encodings becomes computationally expensive as the number of required encodings increases. For instance, let's suppose that a set of N oligos of length n is required such that no member of the set will hybridize with any other member of the set. Let's also assume that all overlaps or shifts of the oligos are to be considered. For each oligo, there are $2n - 1$ shifts or overlaps of at least one base pair. In addition, repetitions of string comparisons are not allowed, and therefore, to insure the condition of no hybridizations $\approx (2n - 1)N^2$ DNA string comparisons have to be made. Therefore, as the size of the set increases, the computational cost increases dramatically. Also, it is difficult to exactly simulate the chemical conditions that determine the hybridization products. Therefore, the application of a DNA based evolution program to the search for good encodings has several possible advantages. The chemical complexity is naturally present, the massive parallelism of the DNA lends itself to demanding problems, and the technique could be implemented *in-situ* to generate specialized sets of encodings for specific problems.

In this paper, an algorithm is proposed that merges DNA based techniques and evolutionary computation. These are successive iteration for better optimization, crossover and mutation operations, and

evaluation and selection operations. The DNA based EP is illustrated with an application to an important problem for DNA computation, the search for good encodings.

DNA Based Evolution Program

The DNA based EP (Figure 3) is implemented with the standard techniques of molecular biotechnology, which include hybridization, ligation, melting and annealing, restriction enzymes, polymerase chain reaction (PCR), nucleases, and repair enzymes[3]. In this paper, it is applied to evolve a good set of DNA encodings. A good set of DNA encodings is a set of oligos of a specified length n that do not mismatch hybridize, hairpin, or shift and hybridize (Figure 2).

The stoichiometric equation for hybridization of two arbitrary oligonucleotides, x_i and x_j , is



where x_jx_i represents the hybridized oligonucleotides. In a DNA computation, there are many reactions like Eq. 1. The direction of the reaction in Eq. 1 is determined by the sign of the change in the Gibb's free energy (G) of the reaction,

$$\Delta G = \Delta G^\circ + RT \log Q. \quad (2)$$

where ΔG° is the free energy change under standard conditions of concentration and pressure, R is the gas constant, and T is the temperature. The concentration factor, Q , is,

$$Q = \frac{[x_i x_j]}{[x_i][x_j]}, \quad (3)$$

where $[]$ indicates mole fractions, and therefore, Q is dimensionless. The reaction will be driven towards chemical equilibrium, where the rates to the left and right of \rightleftharpoons in Eq. 1 are equal, and $\Delta G = 0$. The condition for chemical equilibrium, $\Delta G = 0$, translates to

$$\Delta G^\circ = RT \log K, \quad (4)$$

where K is the equilibrium constant, which is given by

$$K = \frac{[x_i x_j]_{eq}}{[x_i]_{eq}[x_j]_{eq}}, \quad (5)$$

where $[]_{eq}$ indicates the equilibrium mole fractions. The reactions will proceed to a greater or lesser degree

according to the size of their free energy changes, and the relative concentrations of reaction products will be related to the equilibrium constant.

The fitness of a set of DNA encodings is related to the chemical thermodynamics of the hybridization reaction. The enzyme ligase creates the sugar phosphate backbone that strings the bound oligos together into a long, double-stranded DNA molecule (Figure 4). If we assume that the effect of the ligase on the equilibrium reaction thermodynamics is small, oligos will hybridize in concentrations related to the free energy changes of reactions like Eq. 1. Those hybridizations that have a large free energy change will be favored. If a hybridization that does not contribute to the computation, like those in Figure 2, has a large enough free energy change, then, a great many oligos will participate in those reactions, and the result will be either inefficiency or errors in the DNA computer. Let's define those oligos which produce the least unproductive hybridizations as the most fit, and those oligos that have a tendency to form a lot of unproductive hybridizations as less fit. Therefore, we need a mechanism that will destroy the oligos in unproductive hybridizations, and perpetuate those in productive hybridizations.

Evaluation and selection (Figure 5) may be implemented by a hobbled repair mechanism, which is found in cells[3], and specially constructed loops that are placed on the ends of selected oligos[11]. The enzyme, *uvrABC*, detects mismatches in double stranded DNA, and removes 12 base pairs (bp) from one of the strands surrounding the mismatch[3]. Then, a combination of exonucleases is added, which destroys looped molecules with mismatches, and double-stranded molecules without loops. By controlling the concentration of enzyme added at each step, and the time for the enzyme to react, the extent of these reactions can be controlled. This is important because it may be desirable to leave some mismatched molecules to maintain some variation in the population of oligos.

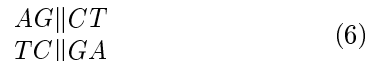
At this stage, there is a choice as to the next step. Over several iterations, the evaluation and selection should result in a very homogeneous population of oligos. In order to conduct a full search of the encoding space, it is necessary to randomize the population. One way of doing this is through a mutagenesis technique (Figure 6). Again, *uvrABC* could be added to remove mismatches, but this time

polymerase is added with it. The effect of the polymerase is to rewrite the sequence in the vicinity of the mismatch to a new, perfectly matched one. Therefore, new sequences can be produced by doing this on some, but not all, of the mismatches. This is done by controlling the enzymatic reaction, through concentration and time, to be incomplete.

In addition, the evaluation and selection step will decrease the concentrations of oligos. To repopulate the oligos, a PCR amplification could be done. This is accomplished by adding a set of random primers, a set of primers corresponding to looped oligos, or both to the population after heating. By heating the looped molecules, the looped strands are separated into circles, and PCR can be done to amplify the concentrations of oligos in the population. The random primers could also be used to create intentional mismatches that *uvrABC* and polymerase would mutate into new oligo encodings.

For efficiency and speed, the last two steps may not be done on every iteration, but only as needed to rebuild the concentrations and to randomize the population.

To recover encodings of the proper length, a restriction site is attached to the ends of each oligo (Figure 7). The purpose of the restriction site is to allow the now double-stranded molecules composed of oligo encodings to be cut up and shuffled, thereby implementing the operation of crossover in an evolutionary program. In addition, the restriction sites are placed so as to recover a set of oligo encodings appropriate for iteration of the DNA based EP. The restriction site could be blunt ended or have an overhang. A blunt ended site would allow greatest variation in how the oligo encodings are shuffled. A choice for a blunt ended restriction site and the enzyme that cuts it is *Alu I* (Figure 7), which has a restriction site,



where $||$ indicates the cut. After cutting at the restriction site, the double-stranded fragments are melted into oligos, and the whole process is repeated (Figure 3). The use of restriction sites in DNA computation techniques was introduced by Head[12] for DNA splicing systems.

Discussion

In the evaluation and selection step, those encodings that produce perfect Watson-Crick complement hybridizations are selected. In one iteration of the DNA based EP, all unproductive hybridizations will not be eliminated. Over several iterations, however, those oligos that are more likely to participate in perfectly matched hybridizations will be favored over those that have a tendency to participate in unproductive hybridizations. Therefore, the probability of selection is related to the equilibrium constant (Eq. 5), and the tendency to participate in perfect or unproductive hybridizations is related to the change in Gibb's free energy for those reactions (ΔG in Eq. 2). Therefore, the larger the free energy gain for a perfect hybridization, the more oligos will be involved in that hybridization through Eq. 5, and the more probable its selection for the next iteration of the DNA based EP.

The DNA based EP could possibly be adapted for application to other problems. One of the issues would be mapping the evaluation function onto the energetics of the DNA hybridization reactions. Evolution programming applications might be well suited to a DNA computer. They could take advantage of the DNA's massive parallelism. Most importantly, they do not require error-free operation. In fact, the mismatched hybridizations could be used to introduce change and variability in the population, and thus, improve the search.

As described elsewhere[11], several of the steps proposed here might also be adapted to implement some problematic and time-consuming operations in DNA computation. If the evaluation and selection operation is iterated and driven to completion, it might be able to eliminate or simplify extraction operations. After enough iterations, it is expected that the only molecules left would be perfectly matched with loops on the ends. The looped molecules could be the starting and ending vertices for the Hamiltonian path. The mutation operation might be adapted for error correction of mismatched hybridizations. In Adleman's original algorithm[2], since he extracted the exact solution for his very simple graph, there was no need for iteration to improve the quality of the solution. For larger problems, however, iteration may become necessary. The number of oligo encodings is limited by the tendency for mismatches to form. Iteration might successively eliminate mismatches, producing

a population consisting of those molecules without mismatches.

Of course, the DNA based EP would have to be verified in the lab, and certainly, would undergo some modification there. The intent, however, was to introduce how DNA computation might be used and done in a different way. If successful, the DNA based EP might be adapted to biotechnology and medical applications. Others[13] have suggested the idea of evolution in a test tube, and this algorithm shows how it might be implemented.

Conclusion

A DNA based EP has been proposed for merging evolutionary search techniques and DNA based computation. An application of the DNA based EP to search for good encodings for DNA based computation was sketched. Finding good encodings for DNA based computation is critical to the technique's success, and is a difficult problem to implement on a traditional computer. Evolutionary programming applications might be well suited for DNA computation, where the DNA's tendency to produce mismatches could be used to expand the range of an EP's search. In addition, the ideas brought forth in the DNA base EP could be adapted to improve specific processes in DNA computation, as well as function as an evolutionary search technique in biotechnology applications.

References

- [1] R. Deaton, R. C. Murphy, M. Garzon, D. R. Franceschetti, and S. E. Stevens Jr., "Good encodings for DNA-based solutions to combinatorial problems," in *Preliminary Proceedings of the Second Annual Meeting on DNA Based Computers* [7], pp. 159–171. DIMACS Proc. Series.
- [2] L. M. Adleman, "Molecular computation of solutions to combinatorial problems," *Science*, vol. 266, pp. 1021–1024, 1994.
- [3] J. D. Watson, N. H. Hopkins, J. W. Roberts, J. A. Steitz, and A. M. Weiner, *Molecular Biology of the Gene*. Menlo Park, CA: The Benjamin/Cummings Publishing Co., Inc, fourth ed., 1987.

- [4] R. J. Lipton, "DNA solution of hard computational problems," *Science*, vol. 268, pp. 542–545, 1995.
- [5] F. Guarnieri, M. Fliss, and C. Bancroft, "Making DNA add," *Science*, vol. 273, pp. 220–223, 1996.
- [6] DIMACS, *Proceedings of the First Annual Meeting on DNA Based Computers*, (Providence, RI), American Mathematical Society, 1996. DIMACS Proc. Series No. 27.
- [7] DIMACS, *Preliminary Proceedings of the Second Annual Meeting on DNA Based Computers*, (Providence, RI), American Mathematical Society, 1997. DIMACS Proc. Series.
- [8] J. Sambrook, E. F. Fritsch, and T. Maniatis, *Molecular Cloning: A Laboratory Manual*. Cold Spring Harbor Laboratory Press, second ed., 1989.
- [9] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*. Berlin: Springer-Verlag, third ed., 1996.
- [10] R. Deaton, R. C. Murphy, M. Garzon, D. R. Franceschetti, and S. E. Stevens Jr., "Genetic search of reliable encodings for dna-based computation," in *Late Breaking Papers*, (Stanford University), First Genetic Programming Conference, 1996.
- [11] R. C. Murphy, R. Deaton, S. E. Stevens Jr., D. R. Franceschetti, J. A. Rose, and M. Garzon, "A new algorithm for DNA based computation," in *Proceedings of the 1997 IEEE International Conference on Evolutionary Computation*, pp. 207–212, IEEE, 1997. Indianapolis, IN, April 13-16.
- [12] T. Head, "Formal language theory and DNA: An analysis of the generative capacity of specific recombination behaviors," *Bull. Math. Biology*, vol. 49, pp. 737–759, 1987.
- [13] S. A. Kauffman, *Origins of Order*. New York: Oxford University Press, 1993.

1. **Begin**

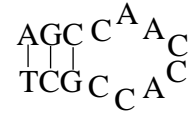
- (a) Initialize Population
- (b) Evaluate Population
- (c) **While Termination Condition Not True**
- (d) **Begin**
 - i. Select New Population
 - ii. Alter New Population with Crossover and Mutation
 - iii. Evaluate New Population
- (e) **End**

2. **End**

Figure 1: Basic Algorithm for an Evolution Program.



Mismatched Hybridization



Hairpin Hybridization



Shifted Hybridization

Figure 2: Hybridizations that produce errors and poor efficiency in a DNA computation.

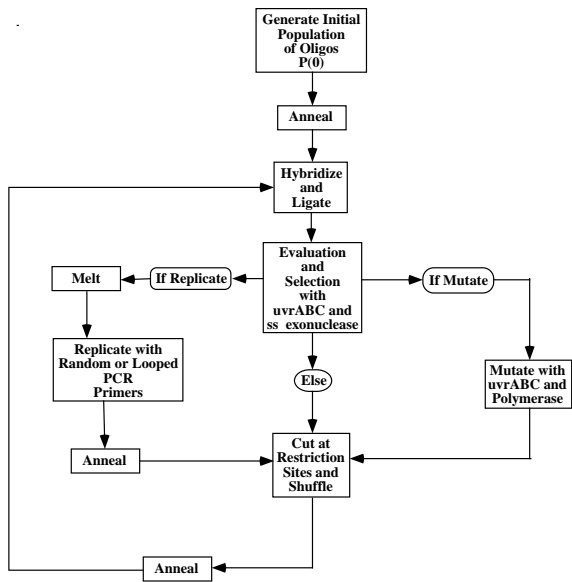


Figure 3: Algorithm for Genetic Search for Good DNA Encodings.

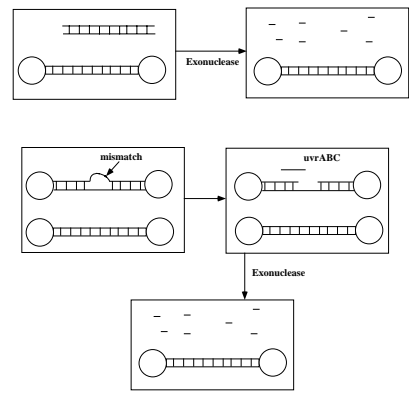


Figure 5: Evaluation and Selection Step.

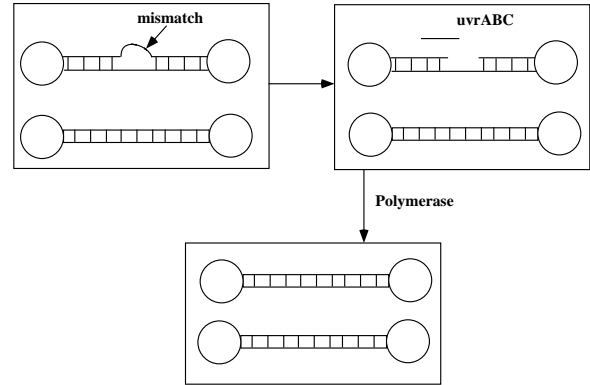


Figure 6: Mutation Step.

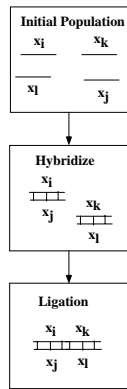


Figure 4: Hybridization and Ligation Step.

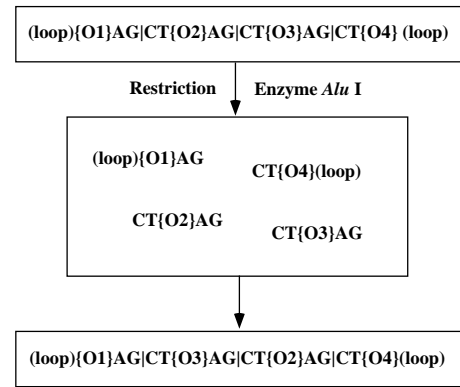


Figure 7: Crossover Step with Restriction Enzyme *Alu I*. For simplicity, only 5' to 3' sequence is shown. *(loop)* stands for loops on ends of molecule, and $\{O1\}$ represents a unique oligo encoding.