Fuzzy logic with biomolecules

R. Deaton, M. Garzon

Abstract The uncertain and inexact nature of the chemical reactions used to implement DNA computations can be turned into an advantage for implementing robust soft computing systems. The key feature of DNA hybridization that makes it appropriate for fuzzy computing is the uncertainty and incompleteness in the formation of a double-stranded duplex from single-stranded oligonucleotides. To implement fuzzy computing, a set of encoding DNA molecules is given that reproduces a specific membership function in the energetics of the DNA duplex. In addition, a fuzzy inference system implemented with DNA hybridization on solid supports is discussed. The ultimate success of this idea as a general technique, however, is dependent on the actual geometry of the Gibbs free-energy landscapes in the space of all duplex formations. Elucidating this problem is undoubtedly of great importance for biomolecular implementation of soft-computing because it may, in particular, shed light on the true import of fuzzy models in biological processes fundamental to life.

1 Introduction

The prototype for a soft computer is a biological organism. To survive, the organism must be able to adapt to changes in its environment, to react flexibly to noise and partial information in its inputs, and to function robustly in the presence of internal limitations, errors, and incomplete information. In an attempt to capture these advantages in computational systems, computer science has pursued biological paradigms, such as fuzzy logic, neural networks, and evolutionary computation. Contrasted with conventional (hard) computing, these types of systems are classified as soft computing. Half a century of research in electronic computing has brought electronic computers to extremely high degrees of reliability when compared to the corresponding biomolecular processes. This property, however, is offset by the brittleness of electronics in adapting to changes in environmental conditions, to respond to noisy and uncertain inputs, and to deal with errors, properties at which biological organisms excel. This fact has even given rise to a new field, evolvable hardware [32, 33], where the methods of evolutionary computation are used in an attempt to develop these abilities in electronics.

Biomolecular computing is a discipline that aims at harnessing individual molecules at nanoscales to perform computations. It makes use of molecular biotechnology to manipulate biological macromolecules for computational purposes. Adleman’s paper [1] established the plausibility of using DNA molecules to solve problems considered difficult for conventional computers, namely, (instances of) NP-complete problems. However, applications of biomolecular computing to conventional computing tasks have turned out to be very sensitive to uncertainty in the reactions between molecules, and the consequent potential for errors in conventional DNA computing has become a paramount issue in the field. Therefore, molecular computing research has moved to address the problem by methods familiar in conventional computing, such as error-correcting codes [22, 8], and novel methods such as error-preventing codes [16, 12].

Conventional soft computing methods, such as fuzzy logic and neural nets, are artificial computational models that exhibit two obvious disadvantages. First, they only capture some abstraction of their biological counterparts, and second, as long as they are implemented in conventional electronics, they inherit the trade-offs mentioned above. A direct implementation of the original paradigms in biomolecules, however, offers what may in fact be the most effective and fundamental solution: take advantage of the uncertainty in the reactions, considered to be a major source of errors in conventional molecular computing, and use it as a source of robustness to change, a feature that characterizes biological organisms. Thus, three established computational paradigms have been implemented in biomolecules, namely, evolutionary algorithms [10, 7], artificial immune systems [11], and neural nets [28, 29].

The purpose of this paper is to explore the feasibility and potential of molecular computing as a substrate for the implementation of fuzzy logic and fuzzy systems. After a brief review of fuzzy logic and fuzzy systems, we describe a DNA implementation of fuzzy membership functions and a fuzzy associative memory is developed. The underlying molecular mechanism, upon which these implementations are based, is DNA hybridization in which two single-stranded DNA molecules (oligonucleotides) bind to form a double-stranded DNA duplex. This reaction is so central to molecular computing that it has been referred to as hybridization logic. As a physical-chemical phenomenon, the hybridization process is inherently fuzzy, i.e., there is a degree of uncertainty as to whether the DNA
molecule is in the double-stranded or single-stranded state. It is this uncertainty in the hybridization that is exploited here to implement fuzzy variables and inferences, and thus, fuzzy hybridization logic. This type of soft computing application for molecular computing takes advantage of the uncertainty in the underlying physical system to implement the computation, rather than attempting to overcome the uncertainty as “error”, as is done in conventional computing.

We will not go into a detailed presentation of molecular computing because there now exist sources in the literature where it can be found. Several recent reviews are recommended for a more detailed introduction to molecular computing, e.g., [13, 17]. Of course, an important source is also the enormous literature on DNA microbiology, an introduction to which can be found in [35]. Similarly, background on fuzzy logic and systems can be found in [24, 23, 9], although we present a brief introduction to the fundamental ideas before the implementation in biomolecules.

2 Biomolecules

Molecules in use in molecular computing are relatively short biomolecules (called oligonucleotides) composed of up to 150 or 200 basic units called nucleotides, the nucleic acid bases [A] (adenine), [G] (guanine), [C] (cytosine), [T] (thymine) (or uraic ([U]) in RNA), that covalently bind to form chains called single-strands, oligonucleotides, or n- mers. Each molecule has a polarity (sense of orientation) from a so-called 5’-end to a 3’-end or vice versa. Oligo- nucleotides can today be synthesized at low costs and manipulated with relative ease despite their nanometric size. Their physical implementation is therefore relatively simple compared to the demanding and costly fabrication processes used in VLSI. Key biotechnology includes gel electrophoresis (for sorting and visualization), cleaving by restriction enzymes (for cut-and-paste), PCR-Polymerase Chain Reaction (for copying), and complex combinations of these basic building blocks. These molecules can be detected, despite their nanometric size, using a number of techniques such as gel electrophoresis, fluorescent labels, and bead separation. They use the fact that DNA molecules are electrically charged (negatively), and absorb less light around the 265 nm range of the spectrum.

In molecular computing, chemical reactions among biological molecules, mainly DNA, DNA analogues, RNA, and various enzymes, are the basic computational mechanism. Perhaps the most important reaction has been the DNA-to-DNA template matching or hybridization reaction, in which two single strands of DNA form a double-stranded DNA helix (Fig. 1). according to the Watson–Crick (herein abbreviated as WC) complement condition, \([A] = [T] (2 \text{ hydrogen bonds}) \) and \([C] = [G] (3 \text{ hydrogen bonds})\), and vice versa. Oligonucleotides bind in an antiparallel way with respect to chemically distinct ends, 5’ and 3’, of the DNA molecule. The primary energetic factor that determines a stable binding is not, however, the obvious hydrogen bonding between individual complementary pairs but the overall energy released in the formation of the helix from a pair of randomly coiled oligonucleotides [5]. As such, this is an ensemble cooperative phenomenon that cannot be localized to a single individual pair, although each plays a role in the final bonding of the two oligonucleotides. Models exist that attempt to understand the primary factors that determine duplex stability. One of them is the nearest-neighbor model that focuses on consecutive base pairs, referred to as base stacks, for which thermodynamic data exists [31].

With approximately \(10^{14}\) hybridizations in Adleman’s original experiment, it is the chemical reactions that give molecular computing its massive search capability. Nevertheless, the lack of methods to control individual chemical reactions causes problems in implementing conventional computing in DNA. For example, most DNA computing models have assumed that only hybridizations that are perfect Watson–Crick complements occur. The reality is that mishybridizations do occur (Fig. 2), i.e. hybridizations that are not perfect Watson–Crick complements, or those that are shifted out of the prescribed binding frame [12, 16]. These mishybridizations can lead to poor efficiency in the molecular assembling during the computation and eventually errors in the computational results. In addition, the degree of completion of enzymatic reactions depends on oligo concentration, salinity, time, and temperature. Therefore, error, imprecision, and uncertainty can be introduced into a DNA-based computation at any step.

3 Fuzzy logic and fuzzy systems

Here we give a brief review of the fundamental concepts of fuzzy logic. Further details on fuzzy logic can be found in [24, 23, 9]. The critical issue addressed by the fuzzy approach is uncertainty. Traditionally, uncertainty has been dealt with through the use of probability spaces and random variables. Probability, however, implicitly assumes

\[
\begin{align*}
5’ \text{AGCTATGATCCGATC} & \quad 3’ \\
3’ \text{TCGATACCTAGGCTA} & \quad 5’
\end{align*}
\]

Fig. 1. Exact Watson–Crick complementary hybridization

\[
\begin{align*}
\text{AGGCTTTAGC} & \quad \text{AGC} \\
\text{TCCAGAAATCG} & \quad \text{C} \quad \text{A} \quad \text{C}
\end{align*}
\]

Mismatched Hybridization

\[
\begin{align*}
\text{AGGCTTTAGC} & \quad \text{AGC} \\
\text{CGAAATCGAA} & \quad \text{C} \quad \text{C} \quad \text{C}
\end{align*}
\]

Hairpin Hybridization

\[
\begin{align*}
\text{AGGCTTTAGC} & \quad \text{AGC} \\
\text{CGAAATCGAA} & \quad \text{C} \quad \text{C} \quad \text{C}
\end{align*}
\]

Shifted Hybridization

Fig. 2. Hybridization mismatches
certainty, since when we toss a fair coin, the outcome is expected to be either heads or tails, both complementary pure events (and hence exclusive of each other). Roughly speaking, probability handles uncertainty by averaging over all possible certain events and creating an imaginary single outcome for the whole set of possibilities. Thus, randomness is far from being uncertainty itself, since the events in a probability space are themselves certain. Each event occurs or it does not, so a particular event itself is not uncertain at all. On the other hand, a 50% chance of hybridization failure between two strands is not the same as hybridization failure in 50% of the strands. After we learn a failure has not occurred, the event of failure sounds like fiction, absolutely nonexistent. As a result, certain basic rules of logic remain valid in probability. For example, the law of the middle excluded has it that an event and its complement cannot both be true simultaneously, and the law of complementarity, that either an event or its complement must always hold. In the example, the outcome must be heads or tails.

Fuzzy sets and logic can be seen as yet another approach to handle uncertainty, one which is entirely different from a probabilistic approach. By contrast, the fundamental assumption of fuzzy logic is that everything possesses all sorts of properties at once, but to a greater or lesser degree, i.e., that everything is a matter of degree. Hybridization of two strands, for example, is an ambiguous event. In general, the strands are partially hybridized and partially not. Both occur to some extent, although, as it were, one of them predominates. The fundamental difference between calculus with ordinary quantities and fuzzy quantities is that fuzzy calculus explicitly declares and manipulates this uncertainty. The basic tenet of fuzzy calculus is that ambiguity exists in a deterministically uncertain fashion. As a result, the old version of certain laws of logic (such as the middle excluded, or the law of complementarity) are no longer true and need to be re-formulated. As paradoxical as it may appear, the outcome is that uncertainty becomes a factor to exploit, rather than overcome. The impact on applications of the models that use a fuzzy approach is an increase in efficiency, responsiveness, and reliability.

The key concepts in fuzzy logic are fuzzy set and fuzzy variable. The concept of a fuzzy set \( H \) presupposes a universe \( X \) all whose members belong to the set \( H \) to some degree, which is given by a so-called membership function.

**Definition 1 (Membership function)** A fuzzy subset \( H \) of a set \( X \) is function

\[ m_H : X \rightarrow [0, 1] \]

where \( X \) is the space, or universal set, and \([0, 1]\) is the real number unit interval.

A fuzzy variable is a variable that takes on fuzzy sets (or fuzzy numbers, as they have been sometimes called) as values. They are usually referred to as linguistic variables. Hybridization \( H \), concentration \( C \), and reliability \( R \) are examples. Fuzzy variables are variables that are quantifiable but not necessarily quantified. That does not mean we that we do not have to define the variable. Fuzzy does not mean sloppy. A variable value needs to be specified so that its values could be completely determined if there were no restrictions on measurement limitations or effort required. In practice, given values are always assumed to be approximate, at best. To handle the variable in the absence of exact values, we estimate the variable in terms of a few linguistic values. For instance, hybridization of two species in a tube can be considered to be WEAK, MEDIUM, or LARGE. Identifying the appropriate variables for a problem is called fuzzification.

Now, fuzzy sets can be operated in a number of ways. A crucial concept is that of a fuzzy relation. Given two fuzzy subsets \( A \) of \( X \) and \( B \) of \( Y \), a fuzzy binary relation \( R \) is a fuzzy subset of the cartesian product \( X \times Y \) such that the membership degree of a pair \((x, y)\) is smaller than both \( A(x) \) and \( B(y) \).

\[ R(x, y) \leq \min\{A(x), B(y)\} \]  \hspace{1cm}  (1)

Fuzzy relations include inferences made on the values of fuzzy variables. They play an important role in fuzzy controllers, to be used below. Fuzzy operators can be used to perform fuzzy inferences and logic. Unlike statistical estimators or traditional artificial intelligence (such as expert systems), fuzzy systems estimate a function without a mathematical formula that gives an explicit description, perhaps simplified, of how outputs depend on inputs. Outputs and inputs are modeled as fuzzy variables. Fuzzy systems store fuzzy associations in terms of common-sense rules, which are either provided by an expert, or observed in an expert’s behavior. For example the dependence between fuzzy variables, hybridization \( H \), concentration \( C \), and reliability \( R \) of a biological protocol can be stated as.

if \( H \) is MEDIUM and \( C \) is MEDIUM, then \( R \) is HIGH

Once fuzzy inferences have been made, a value is calculated for some output fuzzy variable(s). The outcome of a decision process, however, is usually expected in crisp terms. It is then necessary to apply a defuzzification process to convert the value of the output fuzzy variable(s) into crisp values. The process is particularly needed when using a fuzzy model as a control mechanism for a physical device (say, to a fuel injector to control the amount of fuel fed to an engine, or to a heater to keep temperature constant in a system). There are several ways to defuzzify. The most commonly used method consists in taking the average of the range of the output fuzzy variables. In other applications, it may not be necessary to go through a numerical procedure of this sort, particularly, if the crisp valued demanded by the application is directly related to the linguistic values associated with a set of fuzzy variables.

The whole point of this section is now established that, to provide a foundation for fuzzy logic and systems in biomolecules, it suffices to define appropriately the concepts of fuzzy set, fuzzy variable, and fuzzy inference. We turn now to this problem.

4 Fuzzy systems in biomolecules

Baum [2] suggested that a DNA computer based on Adleman’s idea could be used to implement an associative
memory with a capacity larger than that of the human brain. Though never implemented in the lab, the proposal relies on the inherent fuzziness of the DNA hybridization reactions to recall content addressable memories. The idea is that probe memory strands will bind with stored memory strands that are chemically close. Thus, this proposal represents the first connection of DNA molecules with soft computing concepts.

The typical, traditional algorithmic approach (hard computing) specifies an immutable alphabet from which words are composed. Precisely specified algorithms or rules are then used to manipulate or change the words, producing the computational results. The process is exact and precise. Naturally, this was the first approach that was tried when DNA molecules came into computing. Therefore, molecular algorithms were formulated that were supposed to represent the manipulations of molecules in the test tube, and the resulting systems were analyzed for their hard computing power [3, 30]. The basic alphabet was taken to be the nucleic acid bases A, T, G, C from which DNA words were formed to represent problem instances. Rules were implemented with analogs of enzymes or molecular biotechnology. For the most part, however, the rules were over idealized. For instance, the rule for hybridization was duplex formation only in the case of perfect Watson-Crick complements [36]. To take another example, restriction enzymes were expected to cut all DNA double strands at the specified base sequences [21]. As we have seen, these assumptions are rarely valid in practice.

Furthermore, in the early DNA-based computing protocols, it was assumed that the hybridization reaction is a two-state, all or none process. Under a very narrow range of experimental conditions, this might be approximately true. In general, however, hybridization is fuzzy, and in fact, the event of hybridization between two species in a test tube is very much a continuum of outcomes. Experimentally, the properties of the duplex are accessible with a number of analytical techniques, including ultraviolet absorbance, circular dichromism, and nuclear magnetic resonance [4]. Using UV absorbance, the hybridization reaction between two oligonucleotides is characterized by a melting curve (Fig. 3). The DNA duplex absorbs less UV radiation at a wavelength of 260 nm than single-strands. This effect is termed hypochromicity. Therefore, as temperature changes, the degree of hybridization can be measured by the change in UV absorbance [4]. The hypochromicity can be interpreted as either the fraction of single stranded molecules, or as the fraction of unstacked bases in the sample. The last interpretation is probably more physically relevant, as the DNA duplex can “breathe”, possessing both double (stacked) and single (unstacked) regions in the same molecule. Therefore, there are two kinds of “fuzziness” in whether two double-stranded species of, or even single, DNA molecules are hybridized or not.

4.1 DNA-based fuzzy sets
By our foregoing discussion, we have two choices to represent fuzzy variables in biomolecules. We can use the uncertainty of hybridization to represent fuzzy variables at the level of populations, or we can use the uncertainty inherent in hybridization between individual strands. In this paper we explore a combination of both.

In order to define precisely fuzzy set and fuzzy variable representations in a test tube, let X be the set of all possible oligonucleotide pair combinations in a given DNA-based computation. From X (universal set), the subset A of all hybridized pairs is formed. The idea is that A, the set of hybridized oligonucleotide pairs, is a fuzzy set. A species of perfectly hybridized duplex would have all base pairs stacked, and therefore, have a membership value of 1. In physical reality, our set of oligonucleotides, or even a single pair of oligonucleotides, cannot be divided into distinct sets of hybridized and unhybridized species, but each molecule would have a degree of membership in both. Therefore, a fuzzy variable can be implemented as a species of DNA molecules.

4.2 Fuzzy hybridization logic
An architecture for a fuzzy system is shown in Fig. 4 [24]. The system is composed of a parallel set of fuzzy association rules. The rules are of the form

\[(A_i, B_i), \text{ i.e., if } A_i, \text{ then } B_i.\]

In this system, the output is the weighted sum of the outputs of each individual fuzzy rule. To implement the fuzzy system in biomolecules, membership functions, fuzzy rules, weighting factors, and defuzzification must be given molecular counterparts. The idea is to use DNA hybridization, free energy of formation of the duplex \((\Delta G),\)

![Fig. 3. Typical melting curve where \(f_{ss}\) is fraction of single strands](image)

![Fig. 4. Fuzzy system architecture](image)
and the technology of high-density arrays of DNA, or DNA chips. On a DNA chip, different, single-stranded sequences of DNA are attached to different regions of a solid support [6]. The target single-stranded DNA is fluorescently labeled and washed over the probe strands attached to the chip. Target hybridization to probe is visualized and detected optically, and identified by region on the chip.

4.3 Encoding membership functions

As an illustration, we will show how to implement a fuzzy associative memory as applied to a controller for an inverted pendulum [24]. The inputs are the angle of deflection that the pendulum makes with vertical, $\theta$, and the angular velocity. The output control variable is the current supplied to a motor to balance the pendulum in two dimensions. For all three variables, the universe of discourse is some interval in the real line, which is modeled by seven fuzzy values, as shown in Table 1. We use this example to show a DNA implementation of a membership function in the free energy of formation of the DNA duplex, $\Delta G$ [6]. Our goal is just to show how fuzzy logic can be implemented in biomolecular chemistry, and that soft computing is a natural application for molecular computing.

An important problem in molecular computing solutions is the so-called encoding problem, i.e., how to choose a good set of specific strands to implement particular computations. This problem, in its general form, turns out to be very difficult for all of molecular computing [22, 12, 17]. In the context of our fuzzy application, the task amounts to mapping the membership function onto the sequences of the DNA molecules so that their values can be measured experimentally from the physical properties of the duplexes formed under appropriate reaction conditions. It is desirable that the membership function vary continuously with sequence, i.e., that DNA sequences be close when the membership or input values are close. More specifically, the membership function for the input angle, such as shown in Fig. 5, has to be realistically reproduced by some measurable property of the duplexes formed during the computation.

For duplexes formed from oligonucleotides, many properties, such as free energy of formation, hypochromism, and circular dichroism spectra, are determined not only by base composition, but by the sequence [20]. The simplest base sequence dependence arises from nearest-neighbor interactions. In what follows, all sequences are listed, as usual, in the 5’ to 3’ direction. There are 10 Watson–Crick nearest-neighbors [5], namely

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NL</td>
<td>Negative large</td>
</tr>
<tr>
<td>NM</td>
<td>Negative medium</td>
</tr>
<tr>
<td>NS</td>
<td>Negative small</td>
</tr>
<tr>
<td>ZE</td>
<td>Zero</td>
</tr>
<tr>
<td>PS</td>
<td>Positive small</td>
</tr>
<tr>
<td>PM</td>
<td>Positive medium</td>
</tr>
<tr>
<td>PL</td>
<td>Positive large</td>
</tr>
</tbody>
</table>

![Fig. 5. Membership function for positive small (PS) angular input ($\theta$)](image)

AA/TT AT/AT TA/TA AG/CT GA/TC AC/GT CA/TG GC/GC CG/CG GG/CC.

In addition, for short DNA, such as the oligonucleotides used in molecular computing, end effects are significant. To account for end effects, an additional base pair $E/E'$ is introduced [19], and their nearest-neighbor combinations are added to those above.

| $AE'/ET$ | $TE'/EA$ | $CE'/EG$ | $GE'/EC$ |

Thus there are 14 nearest-neighbor pairs with which to characterize duplexes. Because of constraints on composition of the sequence from nearest-neighbor pairs (the number of times a given base precedes another base has to equal the number of times it follows a base [20]), however, these 14 parameters are not independent. Necessary and sufficient constraints for a given set of parameters to define a valid sequence are given by [19]

\[
N_{AA/TT} + N_{AT/AT} + N_{AC/CT} + N_{AG/CT} + N_{AE'/ET} = N_{AA/TT} + N_{TA/TA} + N_{CA/TG} + N_{GA/TG} + N_{EA'TE'}
\]

\[
N_{AG/CT} + N_{GT/AC} + N_{GC/GC} + N_{GG/CC} + N_{GE'/EC} = N_{AG/CT} + N_{TG/CA} + N_{CG/GC} + N_{GG/CC} + N_{EG/CE'},
\]

\]

(2)

where $N_{XY/YX}$ refers to the number of occurrences of the complementary base pair $XY/YX$ in the molecule. In addition, for strands of fixed length, there is an additional constraint [19]. Therefore, for sequence dependent properties of double-stranded oligomers, the number of independent parameters, or sequences in which to express properties, is given by

\[
N_{IS} = N_T - N_C,
\]

(3)

where $N_{IS}$ is the number of independent sequences, $N_T$ is the number of nearest-neighbor pairs, and $N_C$ is the number of constraints. One choice for the independent sequences is given in Table 2 for double-stranded oligomers of varying length [19].

In general, a given molecule can be represented as a linear combination of the independent sequences in Table 2.
The representation of this duplex in terms of the sequences in Table 2 is
\[ s_1 + 2s_2 + s_3 + s_4 + s_5 + s_6 + s_7 + s_8 + s_9 + s_{10} - 5s_{11} - 5s_{12} \]
where \( s_i \) indicate the sequence and \( x_i \) indicates the number of occurrences of that sequence in the molecule. Once the properties of the independent sequences are determined experimentally, then, properties of linear combinations of those sequences are given by linear combinations of the properties,
\[ [\theta] = \sum_{i=1}^{N_{SS}} \theta_i x_i \]

The fuzziness of hybridization is incorporated when inputs hybridize to the membership molecules in Table 3. Inputs are encoded into a spectrum of DNA molecules, and might not be exact matches to the membership molecules. For examples, let’s say our input molecule was \( 3'\text{GCTGTAATCAGGCTG}' \). Its closest match is the membership molecule for 0.2, but the match is not exact. The free energy released would reflect that fact. In addition, even though the membership molecules approximate the shape of the membership function, the encoding task is not complete yet. Inputs to the system would have to be encoded so that they would hybridize to the appropriate membership molecules. In this encoding task, the objective function is given by Eq. (5) since we want to design a molecule that has a specific property value. In addition, the molecule has to satisfy the sequence constraints (Eq. 2), and has to be able to hybridize to the appropriate membership molecules, which means that their sequences have to be similar. This amounts to an integer programming problem for the design of each molecule, which demonstrates once again the difficulties of designing encodings in molecular computing. In general, the encodings are mostly problem specific, have to be designed by trial and error, and have to account for all the complexity of the problem domain and chemical domain of the DNA molecules. Some progress in the design DNA codes has been made [16, 12, 27], but currently, it remains a difficult unsolved problem.

For implementation on a DNA chip, the membership molecules of a particular fuzzy set are encoded in DNA. A collection of these membership function encodings are affixed to a DNA chip (Fig. 6). Therefore, many membership functions can be implemented on one chip, and
evaluations occur in parallel. The input is then encoded and washed over the array. The number of input strands that hybridize to different areas on the chip (different membership functions) determine the membership values. The measure of closeness is their hybridization affinity, which is related to the degree of Watson–Crick complements and the free energy of formation of the hybrid. Perfect matching is not required, and the fuzziness is directly implemented through the fuzziness of the hybridization reaction. For example, in Table 3, the DNA encodings for different membership values are attached to a solid support. Inputs are encoded, allowed to hybridize to the membership molecules on chip, and sensed optically. Concentrations of hybridized pairs on the chip will be determined by the free energy values encoded in the membership molecules through the equilibrium constant,

$$K_{eq} = \exp(-\Delta G/RT) = \frac{C_{hyb}}{C_x C_y},$$

where $T$ is the absolute temperature, $R$ is the gas constant, $C_{hyb}$ is the concentration of the duplex which is a product of a hybridization between two oligonucleotides ($x, y$) whose concentrations are $C_x$ and $C_y$.

4.4 A DNA-based fuzzy associative memory

Now we pursue the foregoing implementation to show how to implement fuzzy inferences. The fuzzy associative memory maps fuzzy sets to fuzzy sets [24]. In the memory, fuzzy associations are pairs of fuzzy values, ($A_i$, $B_i$), representing the inference if $A_i$, then $B_i$. We propose to implement these fuzzy associations in DNA hybridization on a DNA chip in a manner similar to a surface-based approach to molecular solution of the satisfiability problem [15]. In this method, strands are attached to a surface. A set of target strands are washed over the surface where they hybridize to attached strands. Those strands that are hybridized are “marked”, while those that are not are “unmarked”. The “unmarked”, or remaining single strands on the surface are then destroyed by a single-strand exonuclease. The sequences of the molecules encode conjunctive clauses. Therefore, unhybridized regions represent unsatisfied variables in the clause. In this way, the satisfiability of conjunctive clauses encoded in DNA molecules is evaluated.

This method can be modified to implement fuzzy inferences (Fig. 7). To begin with, the degree of hybridization on the surface is not 100%, and therefore, the hybridization mechanism for fuzziness could be incorporated into their method with appropriate encoding. One might even intentionally encode for mismatches to achieve the proper degree of fuzziness. Therefore, the fuzzy values are represented as DNA oligonucleotides, $A_i$ and $B_i$. To implement the rule, the oligonucleotides representing the fuzzy inferences are attached to different regions of a DNA chip. Appropriate inputs ($A_i$ in Fig. 4) are encoded as in the previous section and will hybridize to a greater or lesser degree based upon their hybridization affinity to the fuzzy inference strands attached to the chip. Fluorescently tagged complements of $B_i$ are also added that hybridize to the $B_i$ in the attached inference strands. Those strands that are unsatisfied, are then destroyed by exonuclease, and the output of the satisfied clauses is detected by light output (Fig. 7). The weighting is achieved by adjusting the concentrations of the $B_i$, and the defuzzification is accomplished by averaging the light output. Inferences with multiple fuzzy variables could be implemented in the same way with multiple hybridizations (Fig. 7).

5 Conclusion

In this paper we have shown that it is quite feasible to implement fuzzy logic and fuzzy operations in biomolecules such as DNA. We have provided a very realistic encoding that approximates the membership function in the Gibbs free energy released upon hybridization. This implementation makes more effective use of the properties inherent in DNA hybridization and, moreover, makes fuzzy computing with DNA more tolerant to errors than conventional molecular computing. It provides an encoding of a numerical (membership functions) in the DNA molecules, and so demonstrates the range of potential uses for molecular computing.

Fig. 6. Membership function implemented on a DNA chip for angular input ($\theta$). Each square represents a distinct DNA sequence representation of a fuzzy set value. Membership is implemented by hybridization affinity for prototype strands.

Fig. 7. Hybridization mechanism to implement fuzzy associations on DNA chip. $\overline{A_i}$ indicates the Watson–Crick complement.
The ultimate success of this idea as a general technique, however, is dependent on the actual geometry of the Gibbs free-energy landscapes in the space of all duplex formations. Despite enormous literature in the field, this geometry remains largely unknown. It is part of the so-called encoding problem for molecular computing. Elucidating this problem is undoubtedly of great importance for biomolecular implementation of soft computing and for molecular computing in general. A better understanding of this energy landscape will, in particular, shed light on the true import of fuzzy models in processes fundamental to biological life.

References

29. Mills AP, Tauberfield M, Tauberfield AJ, Yurke B, Platzman PM Experimental Aspects of DNA neural network computation, this volume