

Probabilistic Reasoning with a Bayesian DNA Device Based on Strand Displacement

Iñaki Sainz de Murieta and Alfonso Rodríguez-Patón

Departamento de Inteligencia Artificial,
Universidad Politécnica de Madrid (UPM),
Campus de Montegancedo s/n, Boadilla del Monte 28660 Madrid, Spain
inaki.sainzdemurieta@upm.es, arpaton@fi.upm.es

Abstract. We present a computing model based on the DNA strand displacement technique which performs Bayesian inference. The model will take single stranded DNA as input data, representing the presence or absence of a specific molecular signal (evidence). The program logic encodes the prior probability of a disease and the conditional probability of a signal given the disease playing with a set of different DNA complexes and their ratios. When the input and program molecules interact, they release a different pair of single stranded DNA species whose relative proportion represents the application of Bayes' Law: the conditional probability of the disease given the signal. The models presented in this paper can empower the application of probabilistic reasoning in genetic diagnosis in vitro.

1 Introduction

Since the birth of biomolecular computation in Leonard Adleman's seminal work [2], different applications have been proposed in the literature. The trend of resolving NP-complete problems during the early years of the discipline [11] progressively evolved towards nanotechnology and biomedicine oriented applications, such as genetic diagnosis and drug delivery automata [5,3,1,4].

An important research line emerged taking advantage of the DNA *strand displacement* phenomenon, which in short can be described as follows: a strand A displaces another strand B from a complex $A'B$, due to the higher affinity between A and A' and the greater stability of the duplex AA' . We cite only a few contributions to this extensive topic introduced by Yurke et al. [23], like for example the design of logic gates [19,21,9], DNA automata [22] and theoretical models [6].

The interest in molecular logical inference was reawakened in 2009 with the work presented by the group of Prof. Shapiro [15], where the authors developed an enzyme driven system able to perform autonomously simple logical deductions with DNA molecules. Since then, Rodríguez-Patón et al. [16,17,13] have been working on the design of enzyme free logical inference models that only exploit the DNA strand displacement operation.

With the exception of the work done by the group of Prof. Benenson about stochastic enzymatic reactions [1,4], all the logical models cited above share a common property: they only implement Boolean logic, and thus their output always represent an absolute truth value (true / false, active / inactive, presence / absence, 1 / 0, etc.). None of these deterministic models is able to deal with uncertain knowledge. Other enzyme free models (but not autonomous) have been presented implementing stochastic paradigms [10,24].

Probabilistic reasoning can be used when we want to consider diagnostic accuracy or uncertainty of tests in our clinical decisions (i.e. classic systems like Mycin [20]). With the motivation of designing a model that can process this uncertainty, this article presents a Bayesian biosensor that makes probabilistic reasoning and whose output represents the probability (value between 0 and 1) of a disease. Such type of device can be used to estimate and update the probability of a certain diagnose based in the light of new evidence, i.e., based on the presence or absence of a new specific signal (or set of signals). The DNA sensor device would encode two different probabilities as program data: the conditional probability of the signal given the disease ($P(\text{signal}|\text{disease})$) and the *prior* probability of the disease ($P(\text{disease})$). Then, when the sensor interacts with an input representing the evidence of a signal (its presence or absence), *Bayes' Law* would be autonomously computed by means of strand displacement cascades, releasing a set of DNA species whose ratio encodes the *posterior* probability of the disease given the input ($P(\text{disease}|\text{signal})$).

The rest of the paper is structured as follows: Section 2 includes a brief review of the main concepts in probability theory and Bayesian inference. Section 3 describes how the model encodes the prior and conditional probabilities, as well as the input evidences. Section 4 shows an inference process example that updates the knowledge of a disease applying the Bayes' rule, and how it is implemented by our model. Section 5 discusses in detail the scalability and the mapping of the biological evidences as inputs to the system. Finally, Section 6 summarizes the conclusions and future work.

2 Principles of the Model

Basic concepts of probability theory and Bayesian inference [14,18] used throughout the article are summarized first:

Random variable. A function whose possible values are numerical outcomes of a random phenomenon. It can take different value domains, so that we can talk about continuous, discrete, or Boolean random variables. This paper will focus on boolean random variables, that can take the value true or false with a certain probability. For example, we can talk about the random variable D representing a given disease, which can be present (true) or absent (false).

Logical proposition. A logical formula expressing an assignment between a random variable and one of its potential domain values. Hence, the propositions $D = \textit{present}$ (also denoted as D_1) and $D = \textit{absent}$ (also denoted as D_0) are the possible formulations that can be hypothesized on the random

variable D . Generic propositions of a given variable are denoted with its corresponding lower case letter, for example, $P(d)$ can refer either to $P(D_1)$ or $P(D_0)$.

Probability function. A function P that assigns a probability to each value in the random variable domain (and thus to each potential logical proposition derived from the variable). Building on the above example, we can talk about the probability of D as the tuple $P(d) = \langle P(D_1), P(D_0) \rangle$. The sum of probabilities of all the values of the domain must be equal to 1:

$$P(D_1) + P(D_0) = 1 \quad (1)$$

When this function is defined without any dependence on other random variables, we call it *prior probability*.

Joint probability. Having a set of different propositions, a_1, \dots, a_n , the probability of all of them happening at the same time is defined by the joint probability function, represented as $P(a_1 \wedge \dots \wedge a_n)$ or $P(a_1, \dots, a_n)$.

Conditional probability. This function can be intuitively seen as the degree of belief in a variable after the observation of other variables related to the first. So the conditional probability of a proposition a given b is the probability of a when b is known to occur. It is commonly denoted as $P(a|b)$. Conditional probability can also be expressed as a function of prior and joint probabilities:

$$P(a|b) = \frac{P(a \wedge b)}{P(b)} \quad (2)$$

This formula can be derived into the so called *product rule*:

$$P(a \wedge b) = P(a|b) \cdot P(b) = P(b|a) \cdot P(a) \quad (3)$$

Continuing the above example, when a disease is extensively studied, the probability of a disease d given the signal s is known and expressed as $P(d|s)$. This is also called *posterior probability*.

Conditional independence. Two propositions a and b are conditionally independent when they do not have any dependency relationship. In such case we can rewrite their probabilities as

$$P(a|b) = P(a); \quad P(b|a) = P(b); \quad P(a, b) = P(a) \cdot P(b) \quad (4)$$

Bayes' Law Can be derived from the conditional probability and the product rule formulations, and is stated as follows:

$$P(d|s) = \frac{P(s|d) \cdot P(d)}{P(s)} \quad (5)$$

This rule, together with the property of independence, are key in probabilistic reasoning and allows the establishment of relationships between probabilities and evidences. It allows to update the certainty value of a hypothesis or a diagnosis (prior probability $P(d)$), in the light of new evidence ($P(s)$) and

the signal likelihood ($P(s|d)$), to obtain an “updated” posterior probability ($P(d|s)$).

Assuming we are able to exhaustively estimate all the probabilities concerning the variable D , we can rewrite the law as:

$$P(d|s) = \alpha \cdot P(s|d) \cdot P(d) = \alpha \cdot P(d \wedge s) \quad (6)$$

Since the sum of the probabilities $P(D = present|s)$ and $P(D = absent|s)$ must be equal to 1, we can treat α as a normalization factor.

3 Encoding

Our sensor model aims to implement the product of the probabilities $P(s|d) \cdot P(d) = P(d \wedge s)$ shown in Equation 6. This will be achieved using single stranded DNA in the encoding of the prior probabilities ($P(d)$) and double stranded complexes in the encoding of conditional probabilities ($P(s|d)$). Also the input evidences will need a specific DNA encoding. Details come below:

Encoding Input Evidences

Input evidences are encoded using single stranded DNA. A strand S_1 represents the presence of the signal, while S_0 represents its absence. As we talk about evidences, only one specie can be present at a time: either only input strands S_1 (meaning the signal is present) or S_0 (meaning the signal is not present). This input will tell the sensor that the prior probability of the disease needs to be updated according to the given evidence.

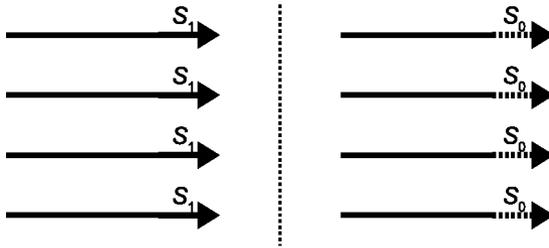


Fig. 1. Encoding input evidences. S_1 represents the presence of the signal; S_0 represents its absence. Only S_1 or S_0 species should be present at the same time.

In case of unwanted presence of one of the two signals in significant concentration, the input could not any more be considered as an evidence as the probabilities P_i ($i = 0, 1$) it would be different of 1 or 0. Thus the computation result would be altered and not valid.

Encoding Prior Probabilities

The prior probability of D is represented as the tuple $P(d) = \langle P(D_1), P(D_0) \rangle$. Our model encodes each value using two different single stranded species: D_1 representing $P(D = present)$ and D_0 representing $P(D = absent)$. The probability values are implicitly encoded in the ratio of molecules of each specie against the total for D . If we denote the number of molecules of each specie A_i as $|A_i|$, we can express the probability as $P(d) = \left\langle \frac{|A_1|}{|A_1|+|A_0|}, \frac{|A_0|}{|A_1|+|A_0|} \right\rangle$. The Figure 2 shows an example DNA encoding of $P(d) = \langle 0.5, 0.5 \rangle$. The coloured toeholds at the 5' end will allow their interaction with the molecules encoding the conditional probabilities.

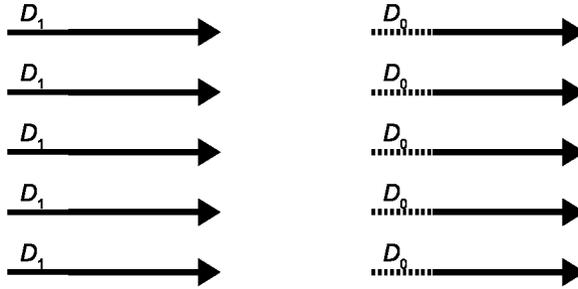


Fig. 2. Encoding prior probabilities. The model encodes each value using two different single stranded species: D_1 representing $P(D = present)$ and D_0 representing $P(D = absent)$. The probability $P(d) = \langle 0.5, 0.5 \rangle$ is encoded as the ratio between the number of molecules of each specie and the total number of species for D .

Encoding Conditional Probabilities

The conditional probability of S given D needs to encode values for the following propositions:

- $(S = present | D = present)$
- $(S = absent | D = present)$
- $(S = present | D = absent)$
- $(S = absent | D = absent)$

The reader can see that for each proposition strand d interacting with a conditional probability molecule, two different outputs encoding two different joint probabilities can be released: $(S = present \wedge d)$ and $(S = absent \wedge d)$. Therefore, from the four different joint probabilities that could be released from the interaction of the D_i species and the conditional probability molecules (representing $(S = present \wedge D = present)$, $(S = absent \wedge D = present)$, $(S = present \wedge D = absent)$ and $(S = absent \wedge D = absent)$), the system needs to be able to select only the outputs corresponding to the input evidence:

- If the input evidence is S_1 , the output strands released should encode ($S = present \wedge D = present$) and ($S = present \wedge D = absent$).
- If the input evidence is S_0 , the output strands released should encode ($S = absent \wedge D = present$) and ($S = absent \wedge D = absent$).

The desired behaviour described above for the conditional probability molecules can be attained using a motif equivalent to the AND gate presented by Seelig et al. [19]. Other motifs implementing such logic could be equally valid, but we have chosen this one due to its simplicity and iteration capability. Figure 3-A shows an example of how the strands building the joint probability $P(S = present \wedge D = present)$ (depicted as $S_1 \wedge D_1$) are released in the presence of the input evidence S_1 and D_1 . Figure 3-B shows the detailed motifs of the molecules that encode the conditional probabilities $P(s|d)$. Similarly to the case of prior probabilities, the probability figures are taken as ratios relating to the number of molecules of each motif. It also shows the formula to establish the correspondence between each motif and the conditional probability values they encode.

In order to ensure we are working with probability values, the following restrictions need to be ensured:

- $\frac{|S_0|D_0|}{|S_0|D_0|+|S_1|D_0|} + \frac{|S_1|D_0|}{|S_0|D_0|+|S_1|D_0|} = 1$ (derived from Equation 1)
- $\frac{|S_0|D_1|}{|S_0|D_1|+|S_1|D_1|} + \frac{|S_1|D_1|}{|S_0|D_1|+|S_1|D_1|} = 1$ (derived from Equation 1)
- $|S_0|D_0| + |S_1|D_0| = |S_0|D_1| + |S_1|D_1|$ (this ensures that different $|S_i|D_j|$ can be mixed in the output for a fixed i)

4 Inference Process

Let us imagine we need to diagnose a disease D with the help of its signal S . The following data is known upfront, due to empirical data:

- Prior probability of the disease:
 - $P(D = present) = 0.5$
 - $P(D = absent) = 0.5$
- Conditional probability of the signal given the disease:
 - $P(S = absent|D = absent) = 0.7$
 - $P(S = present|D = absent) = 0.3$
 - $P(S = absent|D = present) = 0.2$
 - $P(S = present|D = present) = 0.8$

Now we get the confirmation that the signal is present ($S = present$). What is now the probability of the disease being present given that the signal is present, $P(D = present|S = present)$? Since we don't know the prior probability of the signal ($P(s)$), we cannot directly apply the Bayes' Law as stated in Equation 5. We apply the derivation stated in Equation 6 instead:

$$P(D = present|S = present) = \alpha \cdot P(S = present|D = present) \cdot P(D = present) = \alpha \cdot 0.8 \cdot 0.5 = \alpha \cdot 0.4$$

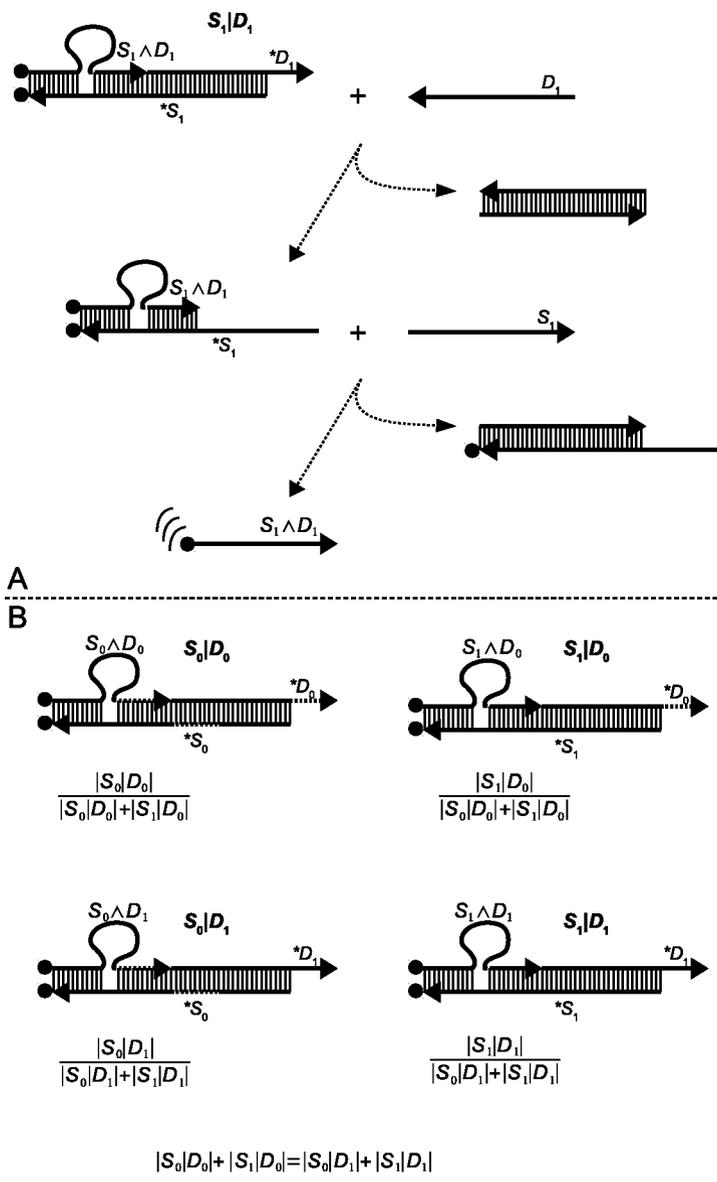


Fig. 3. Encoding conditional probabilities. (A) The top panel shows the hybridization flow releasing the joint probability strand $P(S = present \wedge D = present)$ (depicted as $S_1 \wedge D_1$) when the input evidence $S = present$ (depicted as S_1) and $D = present$ (depicted as D_1). It also releases a fluorophore from its quencher, which will allow the measurement of the output. DNA segments named with an asterisk ($*S_1$ and $*D_1$ in this panel) are Watson-Crick complementary to the corresponding toeholds named without that asterisk. (B) The bottom panel shows the four motifs encoding the conditional probabilities $P(s|d)$, together with the formulas that relate their concentrations to the respective probability values.

In order to find α , we need to calculate $P(D = absent|S = present)$ as well:

$$P(D = absent|S = present) = \alpha \cdot P(S = present|D = absent) \cdot P(D = absent) = \alpha \cdot 0.3 \cdot 0.5 = \alpha \cdot 0.15$$

Since $P(D = present|S = present) + P(D = absent|S = present) = 1$ (see Equation 1) we can derive $\alpha = 1.81$ and $P(D = present|S = present) = 0.73$.

Following the encoding model described in Section 3, we can reproduce with DNA the inference process described above:

- The prior probability $P(d) = \langle 0.5, 0.5 \rangle$ is encoded with two different DNA species, D_1 and D_0 . Each specie will count 50 copies ($|D_1| = 50$, $|D_0| = 50$).
- The conditional probabilities are encoded as follows:
 - $P(S = absent|D = absent)$ is encoded with 70 copies of the complex $S_0|D_0$.
 - $P(S = present|D = absent) = 0.3$ is encoded with 30 copies of the complex $S_1|D_0$.
 - $P(S = absent|D = present) = 0.2$ is encoded with 20 copies of the complex $S_0|D_1$.
 - $P(S = present|D = present) = 0.8$ is encoded with 80 copies of the complex $S_1|D_1$.
- The input evidence $S = present$ is encoded with a unique DNA specie, S_1 , with a number of copies much bigger than the total number of molecules encoding conditional probabilities.

Then the DNA inference process would start by mixing evidences, prior and conditional probabilities all together (see Figure 4):

1. The strands D_j interact with the strands $S_i|D_j$. Assuming an ideal solution (perfectly mixed), the number of complexes $S_i|D_j$ “activated” by strands D_j would be updated as follows:
 - 35 copies of the complex $S_0|D_0$.
 - 15 copies of the complex $S_1|D_0$.
 - 10 copies of the complex $S_0|D_1$.
 - 40 copies of the complex $S_1|D_1$.
2. The input strands S_1 interact with the complexes $S_1|D_0$ and $S_1|D_1$, releasing 15 copies of the strand $S_1 \wedge D_0$ and 40 copies of the strand $S_1 \wedge D_1$.
3. The number of copies of each output strand is estimated by the increase of the different fluorescent colours (red for $S_1 \wedge D_0$ and green for $S_1 \wedge D_1$). The only step missing is the calculation of the probability encoded in that output, which is easily done normalizing both values as follows: $P(D|S) = \left\langle \frac{|S_1|D_1|}{|S_1|D_0|+|S_1|D_1|}, \frac{|S_1|D_0|}{|S_1|D_0|+|S_1|D_1|} \right\rangle = \langle 0.73, 0.27 \rangle$.

5 Discussion

The DNA biosensor presented here operates as a Bayesian inference device, which allows the introduction of quantitative information in the tests, highlighted by the molecular indicators or signals.

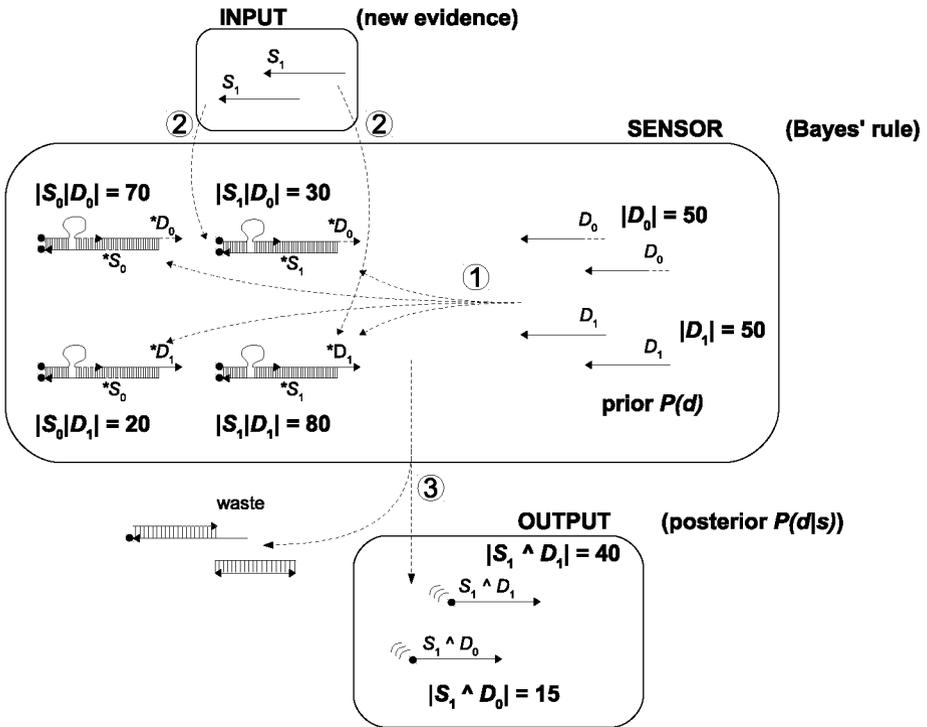


Fig. 4. Inference process. DNA segments named with an asterisk prefix are Watson-Crick complementary to the corresponding toeholds named without that asterisk. (1) The DNA probabilistic reasoning starts with the prior probability species D_i interacting with the conditional probability molecules $S_i|D_j$. (2) Then the input evidence species S_1 interact with the molecules $S_i|D_j$ that had interacted previously with the species D_i , releasing the strand species $S_1 \wedge D_1$ and $S_1 \wedge D_0$. (3) Finally, the probability $P(D|S = present)$ is inferred by normalization of the red and green fluorescence emissions.

Its operation is inspired by the stochasticity of the competing transitions presented by Adar et al.[1], also used by the authors in another work to modulate the ratio between the drug and the drug suppressor in the output [4]. Our work is also based on transitions competing stochastically, but using the DNA strand displacement operation to eliminate the dependence on the enzyme Fok I. Moreover, we have aimed to identify the way to map the basic concepts of probability theory and Bayesian inference and map them into DNA strand displacement motifs, so that they can be used as design patterns when implementing Bayesian reasoning with DNA.

In order this model to have realistic applications in genetic diagnosis, it needs to deal with more than one signal (s^1, \dots, s^n) for the same disease d (superscripts denote the signal number). According to Equation 6, the following formulation of the Bayes' Law would need to be solved: $P(d|s^1, \dots, s^n) = \alpha \cdot P(d) \cdot P(s^1, \dots, s^n|d)$. Assuming conditional independence of the signals given the disease (as in the Naïve Bayes model [12]) we can apply Equation 4 and derive the following expression: $P(d|s^1, \dots, s^n) = \alpha \cdot P(d) \cdot P(s^1|d) \cdot \dots \cdot P(s^n|d)$. We can see that the first two terms of the product (ignoring α) correspond to the formulation we have used for just one signal. Substituting those terms by application of Equation 2 we get this final expression: $P(d|s^1, \dots, s^n) = \alpha \cdot P(s^1 \wedge d) \cdot \dots \cdot P(s^n|d)$. Translating this into our DNA encoding model:

- $P(d)$, $P(s^1)$ and the evidences S_i^1 ($i = 0, 1$) are encoded as described in Sections 3 and 4, with subscripts denoting absence (0) or presence (1) of signal.
- Since the output of the previous steps has the form of species $S_i^1 \wedge D_k$ ($i = 0, 1; k = 0, 1$), the devices encoding $P(s^2|d)$ will need to accept the strands $S_i^1 \wedge D_k$ instead of the strands D_k . The output of this step will release species $S_i^1 \wedge S_j^2 \wedge D_k$.
- Previous step would be repeated for each $P(s^x|d)$ ($x = 1, \dots, n$) until the last signal is reached.

Another important matter to be addressed is the translation of the biological data coming from real samples into the input evidence species. When the evidence to be sensed is determined by the presence of a specific nucleic acid strand, that strand could be directly taken as the input evidence strand S_1 the system expects. The problem comes if the signal is determined by the absence of a specific nucleic acid strand. How can that be mapped into an input evidence strand S_0 ? One possible solution is the addition of an extra “pre-processing” layer, consisting of extra DNA device as described in Figure 5: if no input signal from the samples, the device in the pre-processing layer works as being the input strand S_0 ; but if an input signal is present, it unreleases the strand S_1 . Another potential solution would be the use of a DNA aptamer [25]. Other non DNA aptamers [8] could also be exploited to allow our model take other molecules different from nucleic acids as inputs.

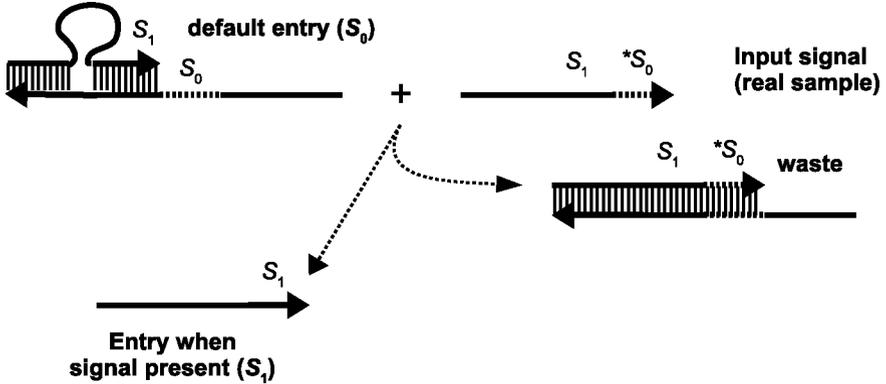


Fig. 5. Processing input as absence of DNA strand. If no signal available among the input samples, the device in the pre-processing layer works as being the input strand S_0 (top-left). However, if an input signal is present (top-right), it unreleases the strand S_1 (bottom-left), leaving a waste molecule that will not react in the system.

6 Conclusions and Future Works

We have introduced a new DNA model for realization of Bayesian inference. The model is completely autonomous, enzyme-free and it is based on DNA strand displacement techniques. Its implementation can be based on experimentally verified and general design derived in [19]. According to the properties examined in [7], the model can be characterized as partly scalable, time-responsive and energy-efficient.

We think the models presented in this paper can empower new quantitative applications of probabilistic genetic diagnosis in vitro. We plan to construct the model in a wet lab and to continue enhancing this model, so that it can be generalized to work with all types of Bayesian networks (and not only the ones following the Naïve Bayes approach [12]).

References

1. Adar, R., Benenson, Y., Linshiz, G., Rosner, A., Tishby, N., Shapiro, E.: Stochastic computing with biomolecular automata. Proceedings of the National Academy of Sciences of the United States of America 101(27), 9960–9965 (2004)

2. Adleman, L.M.: Molecular computation of solutions to combinatorial problems. *Science* 266(5187), 1021–1024 (1994)
3. Benenson, Y., Adar, R., Paz-Elizur, T., Livneh, Z., Shapiro, E.: DNA molecule provides a computing machine with both data and fuel. *Proc. Natl. Acad. Sci. USA* 100(5), 2191–2196 (2003)
4. Benenson, Y., Gil, B., Ben-Dor, U., Adar, R., Shapiro, E.: An autonomous molecular computer for logical control of gene expression. *Nature* 429, 423–429 (2004)
5. Benenson, Y., Paz-Elizur, T., Adar, R., Keinan, E., Livneh, Z., Shapiro, E.: Programmable and autonomous computing machine made of biomolecules. *Nature* 414(6862), 430–434 (2001)
6. Cardelli, L.: Strand Algebras for DNA Computing. In: Deaton, R., Suyama, A. (eds.) *DNA 15*. LNCS, vol. 5877, pp. 12–24. Springer, Heidelberg (2009)
7. Chiniforooshan, E., Doty, D., Kari, L., Seki, S.: Scalable, Time-Responsive, Digital, Energy-Efficient Molecular Circuits Using DNA Strand Displacement. In: Sakakibara, Y., Mi, Y. (eds.) *DNA 16 2010*. LNCS, vol. 6518, pp. 25–36. Springer, Heidelberg (2011)
8. Cho, E.J., Lee, J.W., Ellington, A.D.: Applications of Aptamers as Sensors. *Annual Review of Analytical Chemistry* 2(1), 241–264 (2009)
9. Frezza, B.M., Cockroft, S.L., Ghadiri, M.R.: Modular multi-level circuits from immobilized DNA-based logic gates. *J. Am. Chem. Soc.* 129(48), 14875–14879 (2007)
10. Lim, H.-W., Lee, S.H., Yang, K.-A., Lee, J.Y., Yoo, S.-I., Park, T.H., Zhang, B.-T.: In vitro molecular pattern classification via dna-based weighted-sum operation. *Biosystems* 100(1), 1–7 (2010)
11. Lipton, R.J.: DNA solution of hard computational problems. *Science* 268(5210), 542–545 (1995)
12. Minsky, M.: Steps toward artificial intelligence. *Proceedings of the IRE* 49(1), 8–30 (1961)
13. Sainz de Murieta, I., Rodríguez-Patón, A.: DNA biosensors that reason. *Biosystems* (in press, 2012)
14. Pearl, J.: *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, 1st edn. Morgan Kaufmann (September 1988)
15. Ran, T., Kaplan, S., Shapiro, E.: Molecular implementation of simple logic programs. *Nature Nanotechnology* 4(10), 642–648 (2009)
16. Rodríguez-Patón, A., Larrea, J.M., Sainz de Murieta, I.: Inference with DNA Molecules. In: Calude, C.S., Hagiya, M., Morita, K., Rozenberg, G., Timmis, J. (eds.) *Unconventional Computation*. LNCS, vol. 6079, p. 192. Springer, Heidelberg (2010)
17. Rodríguez-Patón, A., Sainz de Murieta, I., Sosík, P.: Autonomous Resolution Based on DNA Strand Displacement. In: Cardelli, L., Shih, W. (eds.) *DNA 17 2011*. LNCS, vol. 6937, pp. 190–203. Springer, Heidelberg (2011)
18. Russell, S., Norvig, P.: *Artificial Intelligence: A Modern Approach*, 2nd edn. Prentice Hall series in artificial intelligence. Prentice Hall (December 2002)
19. Seelig, G., Soloveichik, D., Zhang, D.Y., Winfree, E.: Enzyme-free nucleic acid logic circuits. *Science* 314(5805), 1585–1588 (2006)
20. Shortliffe, E.H., Buchanan, B.G.: A model of inexact reasoning in medicine. *Mathematical Biosciences* 23(3-4), 351–379 (1975)
21. Soloveichik, D., Seelig, G., Winfree, E.: DNA as a universal substrate for chemical kinetics. *Proceedings of the National Academy of Sciences* 107(12), 5393–5398 (2010)

22. Takahashi, K., Yaegashi, S., Kameda, A., Hagiya, M.: Chain Reaction Systems Based on Loop Dissociation of DNA. In: Carbone, A., Pierce, N.A. (eds.) DNA 11. LNCS, vol. 3892, pp. 347–358. Springer, Heidelberg (2006)
23. Yurke, B., Turberfield, A.J., Mills, A.P., Simmel, F.C., Neumann, J.L.: A DNA-fuelled molecular machine made of DNA. *Nature* 406(6796), 605–608 (2000)
24. Zhang, B.-T., Jang, H.-Y.: A Bayesian Algorithm for In Vitro Molecular Evolution of Pattern Classifiers. In: Ferretti, C., Mauri, G., Zandron, C. (eds.) DNA 2004. LNCS, vol. 3384, pp. 458–467. Springer, Heidelberg (2005)
25. Zhang, D.Y., Winfree, E.: Dynamic allosteric control of noncovalent dna catalysis reactions. *Journal of the American Chemical Society* 130(42), 13921–13926 (2008)