Abstract—Motivated to understand the role of randomness in biological computation, we study a class of urn models that are characterized by urn functions. At each step, one ball is randomly sampled according to an urn function and replaced with a different colored ball. This process is repeated until the urn contains a single color, at which point the process halts. Such an urn can be thought of as a random switch; depending on the initial ball colors and the urn function, the urn population has some probability of converging to any of the initial ball colors. We find that these probabilities have surprisingly simple closed-form solutions and also derive the switching expressions for the time.

We demonstrate the application of such urn models to biological systems by deriving the urn function for the genetic network controlling the lysis-lysogeny decision in the Lambda phage virus. By applying our results to this system, we then derive an intriguing hypothesis on the role of dimers in genetic switches. Many open questions exist on further generalizations of such urn models and their applications to the understanding of randomness and biological computation.

I. INTRODUCTION

A. Randomness in Biology and Computation

The typical engineering mindset is to eliminate randomness – deemed as ‘noise’ – in a system. Yet, randomness has been steadily playing a larger role in computation and problem solving. Beyond the prevalence of Monte Carlo methods that are used to attack high-dimensional problems[2], there have also been surprising theoretical results indicating that randomized methods may, on occasion, be fundamentally more powerful than deterministic algorithms[10][20].

These results are even more thought-provoking when placed next to increasing observations in the biological community of randomness in nature. For example, the firing of neurons is governed by the release of ions into the synapse; these ions arrive in vesicles that have a ‘vesicle release probability’, which has been hypothesized to play a role in several parts of the neural decision making process[19][22].

Many of these high-level behaviors can be traced to biological reactions of the form:

\[ A + B \overset{k}{\rightarrow} C \]

The above equation means that when \( A \) and \( B \) interact with each other, they can be replaced by \( C \) with some probability related to the constant \( k \). Typically, such behavior is analyzed with deterministic or continuous approximations, which are accurate for large numbers of molecules. However, studies show that such approximations can overlook significant system behaviors when small numbers of molecules are involved, such as in genetic networks of single cells[15][3].

One of these behaviors is bistability[21][1], a phenomena where populations of cells or bacteria tend to one of two stable states given the same initial conditions. This behavior is particularly interesting from the perspective of computation because it resembles a switch. A natural question to ask is: how does such switching phenomena arise? How can we design similar mechanisms with a set of randomly interacting agents? An answer to this question would provide a new means for analyzing and designing biological systems and to new approaches for utilizing randomness in computation.

B. Urn models

A natural way to approach this problem is through the literature of urn models. Urn models were mentioned as far back as Bernoulli’s Ars Conjectandi (1713) as a tool for describing different probability distributions. In the model, there is an urn with an initial number of balls, each of which has a color. Balls are then randomly selected with or without replacement to describe different probability distributions.

In the 1900s, researchers began to study the dynamics of urn models under repeated sampling and modification rules. In the general model, a ball is drawn uniformly at random from the urn. If the ball color is \( i \), then \( a_{ij} \) balls of color \( j \) are added into the urn, where \( a_{ij} \) are entries of a matrix. Many results on the convergence properties of such urns have been shown for various classes of urns (see [16] for an excellent survey as well as the classical Johnston and Kotz book[11] and its sequel[13]). For example, in the original Pólya-Eggenberger urn, balls are either red or blue. Each time a ball is sampled, it is returned along with an additional ball of the same color (\( a_{ii} = 1, a_{ij} = 0, i \neq j \)). Letting \( R_t, B_t \) be the numbers of red and blue balls after \( t \) sampling steps and \( \{ X_t, t \geq 0 \} \) be the stochastic process representing the fraction of red balls \( \frac{R_t}{R_t + B_t} \), it was shown that \( X_t \) converges almost surely to the beta distribution \( \beta(R_0, B_0) \).

Urn models have also been generalized in other directions. One particularly interesting model was introduced by Hill et al.[9], who put forward the concept of an urn function. Rather than sampling a red ball with probability \( X_t \) and a blue ball with probability \( 1 - X_t \), they generalize the original Pólya urn by sampling a red ball with probability \( f(X_t) \) and a blue ball with probability \( 1 - f(X_t) \), where \( f : [0, 1] \rightarrow [0, 1] \) is called...
an urn function. The sampled ball is then returned to the urn along with an additional ball of the same color.

C. The Ehrenfest and Mabinogian urns

There are two specific urn models that are of special interest to us. In both of these models, the total number of balls remains constant after each sampling and modification step.

The Ehrenfest urn was introduced in 1907 as a way of modeling diffusion. The urn starts with \( n \) balls, each of which are red or blue. Each time a ball is sampled, it is replaced with a ball of the opposite color (\( a_{ij} = -1 \) and \( a_{ij} = 1, i \neq j \)). If a ball is sampled, it is replaced with a ball of the opposite color. Flajolet and Huillet deal with this by halting the random process at this point. They prove that for an urn with a fixed fraction of red balls halting the random process at this point. They prove that for all \( n \) red or blue balls, there are \( n \) balls of the opposite color to change. Flajolet and Huillet deal with this by halting the random process at this point. They prove that for urns with a fixed fraction of red balls \( c > \frac{1}{2} \), the probability that the urn will converge to all red becomes exponentially close to 1 as \( n \to \infty \).

Each of these urns can be represented by a random walk on the integers \( 0, 1, \ldots, n \). The random walk starts at some initial position \( i \), which represents the number of red balls in the urn. If a ball is sampled, it is replaced with a ball of the opposite color. The probabilities of each of these possible transitions are denoted by \( p_i, q_i, \) and \( r_i \) respectively. When \( p_i = q_i = \frac{1}{2} \) or when \( p_i = p, q_i = q \), there are several standard martingale methods for deriving properties such as absorption probabilities or convergence times of these random walks[4][7]. More recently, Rudolph also derived closed form expressions for many relevant quantities of these random walks with general parameters \( p_i, q_i, \) and \( r_i \).[18]

D. The Model: Fixed size urns with urn functions

The model we define is a Hill-like generalization of the Mabinogian and Ehrenfest urns where sampling is according to an urn function \( f : [0, 1] \to [0, 1] \). At each step,

1) a red ball is sampled \( \text{w.p.} 1 \) \( f(X_t) \),
2) a blue ball is sampled \( \text{w.p.} f(1 - X_t) \),
3) or nothing happens \( \text{w.p.} 1 - f(X_t) - f(1 - X_t) \).

If a ball was sampled, it is replaced with a ball of the opposite color. The urn process halts if it has no more red balls and \( f(0) > 0 \) or if it has no more blue balls and \( f(1) < 1 \).

The main intuition in sampling red balls w.p. \( f(X_t) \) and blue balls w.p. \( f(1 - X_t) \) (as opposed to \( 1 - f(X_t) \)) is to use the urn function \( f \) to capture structure or symmetry in the interactions. Then, the sampling of red and blue balls should also have a symmetry to it. With this definition, we can note that the Ehrenfest urn is simply a fixed size urn with the urn function \( f(p) = p \) while the Mabinogian urn is a fixed size urn with urn function \( f(p) = 1 - p \).

In this paper, we highlight the main results, leaving the details and additional work for the longer version[14]. The remaining sections will be structured as follows:

1) (Section II) We derive the switching probability of first reaching all red or blue. Specifically, given an initial urn configuration with some urn function \( f \), we find the probability that it converges to (or first hits) all red or all blue. This distribution is shaped like the cumulative distribution function of a generalized binomial-like distribution and acts as a soft switch for certain \( f \); that is, if a sufficient majority is red or blue, the urn will converge to the respective majority.

2) (Section III) We show an application of the above results to a proof-of-principle analysis of Lambda phage, a virus that infects the E. Coli bacterium. We find that, with a little work, we are able to describe the lysis-lysogeny decision with an urn function. In addition, we derive a surprising hypothesis on the importance of dimers in genetic switches: if the same negative feedback switch was implemented with monomers, we would no longer get crisp switching behavior.

3) (Section IV) We summarize other results detailed in the long version: a gaussian approximation to the switching probability, the entropy of a switch, and expected switching time.

II. Switching Probability

We are interested in characterizing the switching behavior of a fixed size urn with an arbitrary urn function. Specifically, let \( T \) be the first time when the balls are all red or all blue. We want to find \( \beta_n(i) = \Pr[R_T = n | R_0 = i] \). In this section, we first define a useful intermediate operator \( \mathcal{U}_n \). We then derive \( \mathcal{U}_n f \) for simple functions and prove results on the composition of functions. Finally, we derive switching probabilities of urns in terms of these expressions.

We will use the notation \( \delta_n(i) = \beta_n(i) - \beta_n(i-1) \). Also, \( \tbinom{n}{k} \) will be generalized to allow reals (see Fowler[6] for intuitions), so \( \tbinom{n}{k} = \frac{1}{\Gamma(k+1)\Gamma(n-k+1)} \). We will also have expressions where the notations \( \prod k=1^n (\cdot) = 1 \) and \( \sum k=1^n (\cdot) = 0 \) are assumed.

Definition 1. Define the operator \( \mathcal{U}_n \) as

\[
(\mathcal{U}_n f)_i = \sum_{k=1}^{i-1} \frac{f\left(\frac{1}{n} f\left(\frac{k}{n}\right)\right)}{\binom{n}{k}}
\]

Theorem 1. Let \( f(p) = p - z \), where \( z \in \mathbb{C} \) is a real number such that \( z \neq \frac{k}{n} \) for any \( k = 1, 2, \ldots, n \). Then,

\[
(\mathcal{U}_n f)_i \propto \frac{(2z - 1)n - 1}{(z - 1)n + i - 1}
\]

and

\[
(\mathcal{U}_n f)_i \propto \frac{(1 - 2z)n - 1}{-2n + i - 1}
\]
Proof:

\[(\mathcal{U}_n f)_i = \prod_{k=1}^{i-1} \frac{f\left(\frac{k}{n}\right)}{\frac{k}{n} - \frac{z}{n}} = \prod_{k=1}^{i-1} \frac{k - zn}{n - k - zn}\]

Recall that \(\Gamma(x + 1) = x\Gamma(x)\). Then we have

\[(\mathcal{U}_n f)_i = \frac{\Gamma(i - 1 - zn + 1) / \Gamma(0 - zn + 1)}{\Gamma(n - 1 - zn + 1) / \Gamma(n - i - zn + 1)} \Gamma(i - zn) \Gamma(n - i - zn + 1) \approx (1 - 2zn - 1)^{-1} (zn + i - 1)^{-1}

Alternatively, we also have

\[(\mathcal{U}_n f)_i = \prod_{k=1}^{i-1} \frac{zn - k}{zn - n + k} = \frac{\Gamma(zn - 1 + 1) / \Gamma(zn - i + 1)}{\Gamma(zn - n + 1) / \Gamma(zn - n + 0 + 1)} \Gamma(zn - n + i) \Gamma(zn - i + 1) \approx (2z - n - 1)^{-1} (zn - n + i - 1)^{-1}

Theorem 2. Let \(f(p) = g(p)h(p)\). Then,

\[(\mathcal{U}_n f)_i = (\mathcal{U}_n g)_i \cdot (\mathcal{U}_n h)_i\]

Proof: By applying Definition 1,

\[(\mathcal{U}_n f)_i = \prod_{k=1}^{i-1} g\left(\frac{k}{n}\right) h\left(\frac{k}{n}\right) = (\mathcal{U}_n g)_i \cdot (\mathcal{U}_n h)_i\]

Theorem 3. Let \(f(p) = \frac{g(p)}{g(p) + q(1-p)}\). Then,

\[(\mathcal{U}_n f)_i = (\mathcal{U}_n g)_i\]

Proof: By applying Definition 1,

\[(\mathcal{U}_n f)_i = \prod_{k=1}^{i-1} \frac{g\left(\frac{k}{n}\right)}{g\left(\frac{k}{n}\right) + g\left(\frac{n-k}{n}\right)} = \prod_{k=1}^{i-1} \frac{g\left(\frac{k}{n}\right)}{g\left(\frac{n-k}{n}\right)} = (\mathcal{U}_n g)_i\]

We can now derive the switching probabilities in terms of \(\mathcal{U}_n f\). We will express the results in terms of \(\delta_n(i)\). Note that since \(\beta_n(0) = 0\), we have \(\beta_n(i) = \sum_{j=1}^{i} \delta_n(j)\).

Lemma 1. For a random walk between absorbing boundaries 0 and \(n\), and general transition probabilities \(p_i, q_i\), and \(r_i\) for transitions \(i \rightarrow i + 1, i \rightarrow i - 1,\) and \(i \rightarrow i\), we have

\[\delta_n(i) = \frac{1}{Z} \prod_{k=1}^{i-1} \frac{q_k}{p_k}\]

where \(Z = \sum_{j=1}^{n} \prod_{k=1}^{j-1} \frac{q_k}{p_k}\).

Proof: \(\beta_n(i)\) satisfies the recurrence relation

\[\beta_n(i) = p_i \beta_n(i + 1) + q_i \beta_n(i - 1) + r_i \beta_n(i)\]

Then using the definition of \(\delta_n(i)\), we can simplify this in the following way:

\[(p_i + q_i + r_i) \beta_n(i) = p_i \beta_n(i + 1) + q_i \beta_n(i - 1) + r_i \beta_n(i)\]

\[\delta_n(i + 1) = \frac{q_i}{p_i} \delta_n(i)\]

\[\delta_n(i + 1) = \delta_n(1) \prod_{k=1}^{i} \frac{q_k}{p_k}\]

Then we get our final result by noting that \(\beta_n(n) = 1\).

Theorem 4. Let a fixed size urn sample a red ball with probability \(k_r f(X_t)\) and a blue ball with probability \(k_b f(1 - X_t)\), where \(0 \leq k_r f(x) + k_b f(1 - x) \leq 1\) for \(0 \leq x \leq 1\). Then,

\[\delta_n(i) = \frac{1}{Z} \prod_{k=1}^{i-1} \frac{k_r f\left(\frac{k}{n}\right)}{k_b f\left(\frac{n-k}{n}\right)} = \frac{1}{Z} \left(\frac{k_r}{k_b}\right)^{i-1} (\mathcal{U}_n f)_i\]

Proof: By Lemma 1 and Definition 1, we have

\[\delta_n(i) = \frac{1}{Z} \prod_{k=1}^{i-1} \frac{k_r f\left(\frac{k}{n}\right)}{k_b f\left(\frac{n-k}{n}\right)} = \frac{1}{Z} \left(\frac{k_r}{k_b}\right)^{i-1} (\mathcal{U}_n f)_i\]

III. AN ANALYSIS OF A BIOLOGICAL SWITCH

A. Lambda phage Background

The behavior of Lambda phage is controlled by a small ‘molecular program’ written into its DNA. When Lambda phage injects its DNA into the host cell, the host will transcribe proteins encoded by the DNA. These proteins can interact with each other and also affect the future transcription of other proteins. The primary initial decision that Lambda phage needs to make is whether to go into the lytic or the lysogenic cycle.

If the host cell is healthy, then Lambda phage would rather enter the lytic cycle where it repeatedly duplicates itself until it bursts the cell and finds another host to infect. However, if the host cell is unhealthy, or if there are already other infections in the bacterium, then it is likely that there will not be many other host cells to infect. In this case, Lambda phage prefers to enter the lysogenic cycle where it remains dormant until a better time.

The genetic circuits of Lambda phage have been thoroughly studied (see [17] for more details). Roughly speaking, there is a portion of the network that is in charge of sensing environmental signals and setting two proteins, cro and CI, at certain concentration levels. As the levels of cro and CI increase, they inhibit the previous proteins, which slowly degrade away.

The main decision making portion now ends with a biological switch implemented by the cro and CI proteins. It can be described in three parts:
1) The cro and CI proteins can combine to form dimers according to cro + cro → cro₂ and CI + CI → CI₂.

2) The dimers can bind to a promoter region on the Lambda phage DNA; the monomers can be degraded at some rate.

3) When RNA polymerase (RNAP) attaches to transcribe (create) a new protein, it will transcribe a cro protein if it moves right along the DNA and a CI protein if it moves left along the DNA; however, if a cro dimer is bound, it will block RNAP from transcribing CI. Likewise, if a CI dimer is bound, it will block RNAP from transcribing cro².

The system will tend to either all cro proteins or all CI proteins since whichever protein begins to dominate will block further transcription of the opposing protein. Existing proteins of the opposing type will gradually degrade over time according to the degradation rates of the cro and CI monomers. Eventually, if cro dominates, then Lambda phage will enter the lytic cycle. If CI dominates, it will enter the lysogenic cycle.

B. Modeling the Lambda Phage Switch

Because of the small numbers of proteins involved, deterministic modeling of this system does not accurately predict experimental results and stochastic simulations are the primary method of analyzing such systems[1]. We will show how the behavior of the Lambda phage switch can be modeled with a fixed-size urn with an urn function.

In our model, we begin the urn process from the time when the other proteins are no longer relevant to the continued production of cro and CI and have reached a total number of proteins where the degradation rate is approximately equal to the transcription rate (so the number of proteins remains roughly constant). In addition,

1) We approximate transcription and degradation as occurring simultaneously.

2) We assume the rate of the reversible monomer-dimer reaction occurs much faster than the rate of transcription, so that the relative dimer-monomer concentrations can be approximated as in equilibrium.

3) We assume that most of the time, one of cro₂ or CI₂ is bound to the promoter.

It is now easy to see how such a system fits into our model: the cro and CI proteins are the red and blue balls in the urn. For p = \#cro/\#cro + \#CI, let d_{cro}(p) and d_{CI}(p) be the probabilities that either cro or CI, respectively, will be the next protein to degrade. Let t_{cro}(p) and t_{CI}(p) be the probabilities that either cro or CI, respectively, will be the next protein to be transcribed. Then, for i representing the number of cro proteins, we can calculate the probability of replacing a cro with CI or vice versa. Specifically, for the corresponding random walk on a line, q_i = d_{cro}(\frac{i}{n})t_{CI}(\frac{i}{n}) and p_i = d_{CI}(\frac{i}{n})t_{cro}(\frac{i}{n}).

For the following expressions, we will use K₁ and K₂ for the equilibrium constants of cro and CI dimer formation respectively, D₁ and D₂ as the degradation rates of cro and CI respectively, E₁ and E₂ as constants calculated from the binding energies of cro and CI to the DNA, n as the total number of cro and CI in solution, and [n] as the concentration of n in moles per liter, e.g. [n] = moles \cdot n = \frac{\text{mole}}{V \cdot nₐ}, where V is the volume and Nₐ is Avogadro’s number.

The first observation to make is that despite what we have claimed previously, it is not clear that this system can be modeled by an urn function. Because of the reversible dimer formation reactions, it is not true that d_{cro\,\text{d}} and t_{cro\,\text{d}} are solely functions of p = \frac{i}{n}, the fraction of cro proteins. Indeed, as seen in Lemma 2 below, the expressions are also dependent on the volume of the cell, as captured by [n].

Lemma 2.

\[ d_{cro}(p) \propto D₁(\sqrt{8K₁[n]p + 1 - 1})/K₁ \]
\[ d_{CI}(p) \propto D₂(\sqrt{8K₂[n](1 - p) + 1 - 1})/K₂ \]
\[ t_{cro}(p) \propto E₁(\sqrt{8K₁[n](p + 1 - 1)²}/K₁ \]
\[ t_{CI}(p) \propto E₂(\sqrt{8K₂[n](1 - p) + 1 - 1)²}/K₂ \]

Proof: d_{cro,\text{d}}(p) is proportional to the number of cro and CI monomers respectively multiplied by the degradation rates; t_{cro,\text{d}}(p) is proportional to the number of cro₂ and CI₂ dimers multiplied by the binding probabilities. The proof is detailed in the long version.

Theorem 5 (Probability of Entering the Lytic Cycle). For an E. Coli bacterium infected by a single Lambda phage virus, let n be the total number of cro and CI proteins when the other (CI, CIII, N) protein concentrations have become negligible. Represent the number of cro proteins with i and the number of CI proteins with p − i. Then for small concentrations of protein [n] → 0, the probability of entering the lytic cycle can be found by

\[ \delta_n(j) = \frac{1}{Z} \left( \frac{D₁E₂K₂}{D₂E₁K₁}\right)^{j - 1} \left( \frac{n - 1}{j - 1} \right) \]

Similarly, for large concentrations of protein [n] → ∞,

\[ \delta_n(j) = \frac{1}{Z} \left( \frac{D₁E₂\sqrt{K₂}}{D₂E₁\sqrt{K₁}}\right)^{j - 1} \left( \frac{n - 1}{j - 1} \right)^{1/2} \]

C. The Role of Dimers

The previous result becomes even more interesting when we consider what happens without dimers. Consider the following situation: suppose that the CI and cro monomers did not form dimers, but could directly bind to the promoter region to suppress transcription of the opposing protein. What would happen then? This question is essentially asking whether the negative feedback of the switch is sufficient for creating switch behavior. Surprisingly, we find that in this case, we do not get crisp switching behavior anymore (see Fig. 1).

Theorem 6. For an E. Coli bacterium infected by a single Lambda phage virus, let n be the total number of cro and CI proteins when the other (CI, CIII, N) protein concentrations have become negligible. Suppose that cro and CI proteins
B. Expected switching time

It is also important to characterize the time it takes to switch. In the longer version, we derive bounds on the expected switching time for fixed size urns which sample red balls with probability \( f(X_i) \) and blue balls with probability \( f(1 - X_i) \).

C. Next steps

There are several directions for extending this work. On the model itself, it would be great to generalize the results to multi-colored urns (with multivariable urn functions) or to allow more types of transitions. It would also be interesting to use urn functions to analyze other behaviors, such as oscillations. Finally, it would be interesting to see what other algorithmic applications may arise from fixed size urns with urn functions.

ACKNOWLEDGMENT

The authors would like to thank Tony Wong for helpful discussions on the switching probability and Persi Diaconis for pointing us to the Pólya urn literature. This work was supported in part by the Molecular Programming Project funded by the NSF Expeditions in Computing Program under grant CCF-0832824.

REFERENCES