Improved optical multiplexing with temporal DNA barcodes

Shalin Shah,[†] Abhishek K Dubey,[‡,¶] and John Reif[*,†,‡]

†*Department of Electrical & Computer Engineering, Duke University, NC, US - 27701*

‡*Department of Computer Science, Duke University, NC, US - 27701*

¶*Computational Sciences and Engineering Division, Health Data Sciences Institute, Oak Ridge National Lab, TN, US - 37831*

E-mail: reif@cs.duke.edu

Abstract

Many biochemical events of importance are complex and dynamic. Fluorescence microscopy offers a versatile solution to study the dynamics of biology at the mesoscale. An important challenge in the field is the simultaneous study of several objects of interest, referred to as *optical multiplexing*. For improved multiplexing, some prior techniques used repeated reporter washing or the geometry of nanostructures; however, these techniques may require complex nanostructure assembly, multiple reporters or advanced multistep drift correction. Here we propose a time-based approach, for improved optical multiplexing, that uses readily available inexpensive reporters and requires minimal preparation efforts. We program short DNA strands, referred hereby as *DNA devices*, such that they undergo unique conformation changes in presence of the dye-labeled reporters. The *universal fluorescent reporter* transiently binds with the devices to report their activity. Since each device is programmed to exhibit different hybridization kinetics, their fluorescent time trace, referred to as the *temporal barcode*, will be unique. We model our devices using Continuous-time Markov Chains

and use stochastic simulation algorithm to generate their temporal patterns. We first ran several simulation experiments with a small number of our devices, demonstrating several distinct temporal barcodes, all of which use a single dye color. Later, using a nanostructure, we designed a much larger pool of unique temporal barcodes and performed supervised learning using support vector machine. Our simulation experiments and design principles can aid and influence the experimental design of temporal DNA barcodes.

Keywords: DNA kinetics, Optical multiplexing, Temporal patterns, Machine learning, Single-molecule imaging, Continuous-time Markov chain

Introduction

Far-field microscopy can offer a dynamic view of biology at the cellular and molecular scale. In the past decade, single-molecule localization microscopy (SMLM) techniques[1,2] have made tremendous progress. These techniques achieve sub-diffraction imaging resolution by temporally separating the fluorescence of the targets closer than the diffraction limit. This is achieved by switching the target between the fluorescent-ON and the fluorescent-OFF states for which several different ways have been proposed.[3–5] The temporal separation is used to find a centroid of each fluorescent spot separately and achieve sub-diffraction resolution by a manual overlay of image stack with localizations. Multiplexed imaging with more than one dye color have been incorporated into existing techniques to push the resolution limit further.[6]

Further multiplexing beyond the three basic colors is desirable therefore a few different multiplexing techniques have been proposed.[7–19] The simplest form is wavelength multiplexing where several fluorescent colors are used. The state-of-the-art wavelength multiplexing technique by Woehrstein et al. uses a DNA nanostructure to demonstrate over 100 colors by the linear combination of simple RGB dyes.[20] Such techniques are easy to implement in practice, however, they are fundamentally limited by the number of non-overlapping dye

emission spectra. Another class of multiplexing techniques includes using the geometry of a mesoscale structure such as the DNA origami barcodes proposed by Lin et al. [7] These uniquely identifiable nanostructures can later be used as a taggant for studying the species of interest. These techniques generally have much more room to encode information, however, they require complex nanostructure self-assembly.

Passive encoding techniques such as Exchange-PAINT[21,22] have also been reported. We refer to such techniques as passive since the multiplexing information is not directly visible as different fluorescent colors. State-of-the-art technique Exchange-PAINT uses the sequence of fluorescent reporters to encode the multiplexing information. A given fluorescent DNA reporter can only report a subset of locations that are complementary to it. Once these locations are successfully reported, the current reporters can be washed out of the sample chamber for a new set of fluorescent DNA strands which can then report new locations or sample types. This process can be repeated a few times until all the required sample species have been successfully reported to achieve high optical multiplexing using a single dye color but multiple fluorescent DNA strands. Although the multiplexing capacity of Exchange-PAINT is theoretically infinite, it requires multiple dye-labeled reporter strands and washing steps, which makes it slightly expensive to implement, requires a complex hardware setup and, finally, needs multistep drift corrections to account for the additional drift due to washing.

In this work, we introduce a generalized time-based reporting framework to improve the multiplexing capabilities of the objects studied using fluorescence microscopy systems. Our framework contains a set of short DNA strands called *DNA devices* and a complementary universal fluorescent reporter strand called *reporter*. These DNA devices can be attached to the glass surface and observed using total internal reflection fluorescence microscopy (TIRFM)[23] since this will limit the background fluorescence of the reporter strands. Each time a reporter transiently attaches to our DNA devices it comes in the focal plane of the camera and we see a short-lived bright fluorescent spot. Since each device is programmed to

undergo unique conformational changes, the stochastic intensity trace of each DNA device will be distinct, if captured for sufficient time. We define these fluorescent intensity traces as *temporal DNA barcode* since they can uniquely identify the underlining DNA device. We model our devices using time-homogeneous Continuous-time Markov Chains (CTMC) and use stochastic simulation algorithm (SSA) to generate their temporal patterns.

There have been some prior works such as hairpin-based nanoclocks,[24] nanoparticle doping distances,[17] resonance energy transfer (RET) based temporal taggants[25] which use the time-domain to encode information. However, to our knowledge, our work is the first to introduce the use of temporal signatures and DNA hybridization kinetics for the unique identification of single-molecules. Several other sophisticated applications also exploit the programmable nature of DNA hybridization[26–35] making the time ripe for the development of frameworks such as ours that require tuning the DNA hybridization kinetics. Our work also provides a systemic approach to searching a set of DNA devices from the design space such that they follow experimental restrictions.

The rest of the paper is organized as follows. First, we introduce the abstract modeling of simple DNA devices using the time-homogeneous CTMC. Second, we use the stochastic model to predict a set of programmable parameters to generate a large pool of distinguishable DNA devices. In particular, we program the following parameters: (a) the length of DNA, (b) the number of available domains, (c) domain sequestering using DNA hairpins, and (d) the competing secondary structure. Third, using these simple devices and a DNA nanostructure as a platform, the temporal barcodes of several devices are combined to achieve a much larger pool of temporal barcodes. Finally, we demonstrate an end-to-end large-scale simulation experiment using a DNA nanostructure with five staple extensions, each of which can act as a DNA device to create a set of 56 different temporal DNA barcodes using the principle of linear combination of DNA devices. We perform supervised machine learning by training a classification model and identify the temporal barcodes of designed nanostructure-based devices using the trained model with high accuracy. We close the paper with a discussion

on some potential applications of our technique and opportunities to scale up the temporal barcode set by using multiple dyes.

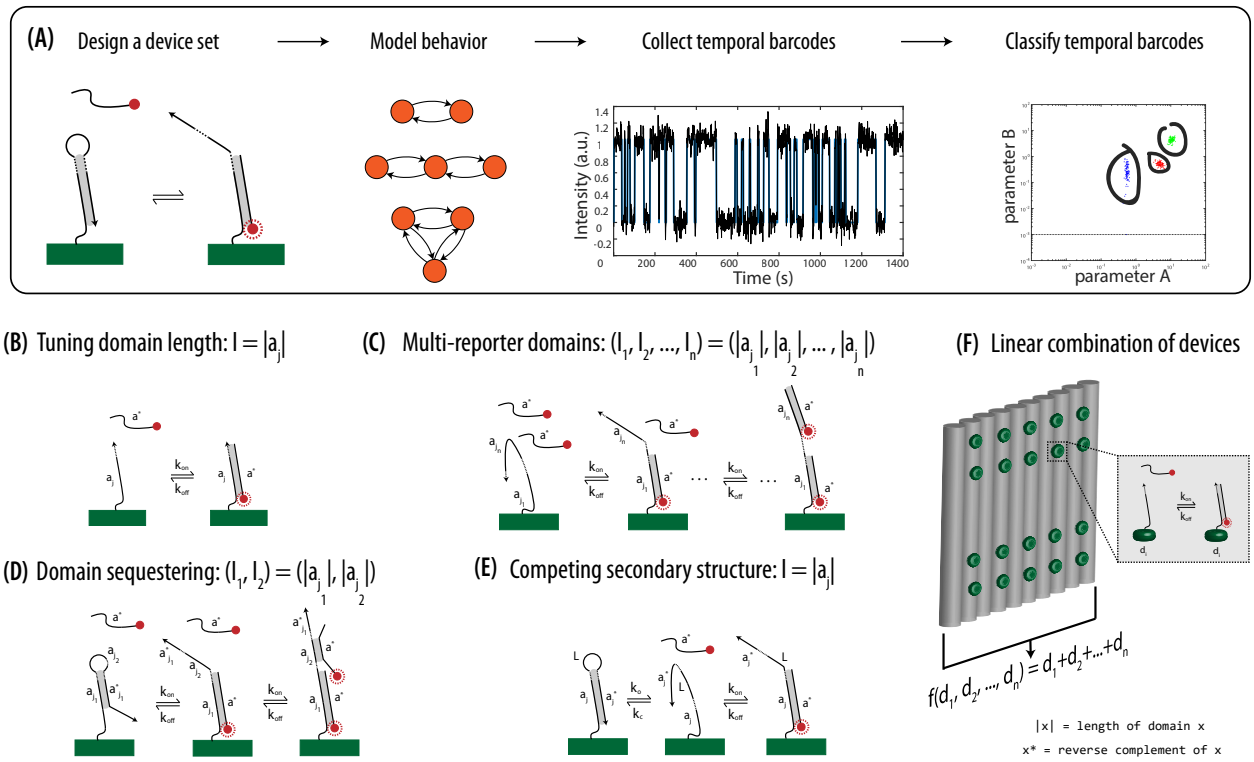

Figure 1: A summary figure illustrating our temporal barcoding framework, with example DNA devices parametrized by domain lengths. (a) The workflow for unique identification of single-molecules using our devices. A set of devices are designed, modeled and simulated to generate temporal barcodes which are analyzed in the parameter space for clustering (or classification). (b) An example DNA device tuning domain length parameter to program the barcode behavior. (c) Designing the number and lengths of reportable domains to tune the temporal barcode. (d) Sequestering a domain to enforce event sequence. (e) Programming length of a competing secondary structure to tune dark-time of the temporal barcode. (f) Using a nanostructure as a breadboard for tuning the linear combination of individual temporal DNA barcodes. Note that only one universal fluorescent is used for all the devices.

Results

Stochastic modeling of DNA devices

The stochastic behavior of DNA hybridization for single-molecule system is often modeled as a time-homogeneous CTMC.[36,37] This way of modeling single-molecule interactions can offer

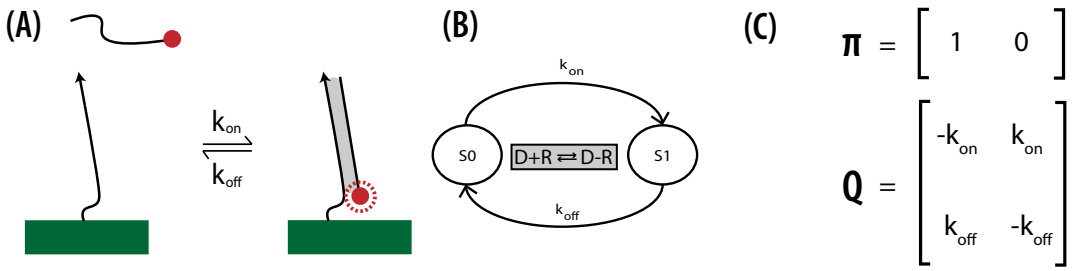

Figure 2: A simple example illustrating the idea of modeling a single-molecule DNA device. (a) A short ssDNA device is attached on a glass surface. When fluorescent DNA strand attaches to the device, it gets reported. (b) A CTMC model showing this transient binding process. (c) The initial probability value matrix and the transition matrix of the 2-state CTMC model. This model can be simulated to generate a state trajectory resembling the ideal fluorescence intensity trace of a single device-reporter combination.

the benefit of adding an abstraction layer to the DNA sequence level, making the simulation process simple yet retaining the necessary details. By modeling DNA hybridization and de-hybridization as a CTMC, we assume that the holding time of each state is an exponentially distributed random variable. CTMC is a stochastic modeling technique where each random variable follows Markov property *i.e.* the probability of the random variable being in the current state depends only on the previous state. Formally, CTMC is a random process $X(t)_{(t>0)}$ with a finite state space S, such that the generated state sequence at time $t+1$ follows the property $P(X_{t+1}|X_t, ...X_2, X_1) = P(X_{t+1}|X_t)$. At any time t, a CTMC is represented using a transition rate matrix Q and a state space S for a given initial probability vector π. The holding time in each state is an exponentially distributed random variable with the rate value λ_{ij}. The CTMC models used in this work can be easily implemented at the corresponding DNA sequence level with existing tools such as NUPACK.[38] Several techniques have been suggested to simulate a CTMC, however, in this work, we will adhere to the stochastic simulation algorithm (SSA) by Gillespie since it is the computationally preferred choice for simulating a few molecules.[39] A simple single-stranded DNA (ssDNA) device and its corresponding CTMC model is shown in Fig. 2. Refer Trivedi, K. (2006)[40] for more details on CTMC.

Programmable DNA device parameters

DNA devices can be designed with several programmable parameters, such as the length of DNA, salt concentration, temperature, and secondary structures. We assume that the experimental conditions such as the temperature, salt concentration and others are kept constant, and only tune the DNA device parameters in this study. All the rate constants of reactions were adopted from prior experimental studies.[37]

Tuning the lengths of binding sequence to create diversity

The simplest way to tune the kinetics of DNA hybridization to modify its length as studies have shown exponential decrease in the melting temperature of double-stranded DNA (ds-DNA) with increased length.[41] Since we want to achieve transient binding of DNA strands, we designed the device domain length in the 7 - 10 nt range. Prior PAINT studies have used these lengths in their single molecule experiments to achieve transient binding behavior using DNA.[4,21,22,37] This length range is short enough to achieve transient DNA hybridization and yet long enough to be captured by the current detector technology. Some single-molecule studies have captured kinetics of shorter DNA strands, however, such probes were noise-free and expensive since they used fluorescence-quencher pairs.[36] Similarly, longer DNA strands can be used however with their average binding time ranging in the several seconds range, the relative immunity to photo-bleaching might be lost. Hybridization and de-hybridization rate constants were adopted from Jungmann et al. (2010).[37]

We represent a DNA device using the notation $|a|$ where $|a| \in S = \{x | x \in [7, 8, 9, 10]\}$ and $|a|$ indicates the length of the domain a. To design a set of temporal DNA barcodes, we modeled the DNA hybridization behavior using a two state CTMC as shown in Fig. 3b where the k_{off} rate for each length will be different. We started with three simulation experiments one for 8 nt, one for 9 nt and one for 10 nt device. The length of simulation experiment was also altered from 10 minutes to 1 hour assuming the complementary fluorescent reporter strand has a concentration of 25 nM. More details about MATLAB scripts, simulation techniques and

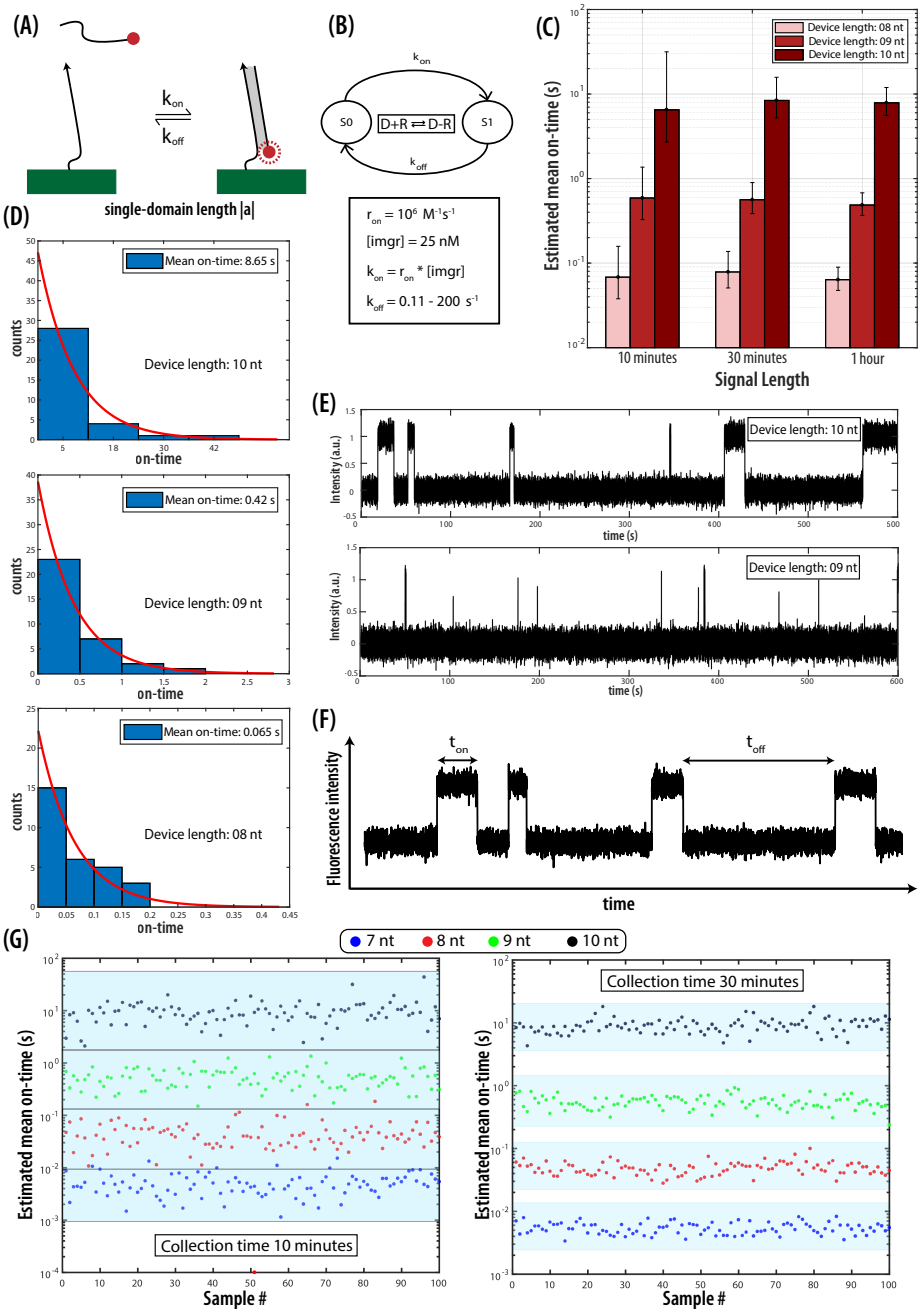

Figure 3: Tuning device length to program temporal barcodes. (a) Transiently binding fluorescent strand to ssDNA device. (b) A 2-state CTMC model representation. (c) Estimated mean value for the generated exponential distributions where error bars indicate a 95% confidence interval. (d) Histograms of on-time for ssDNA devices with length 10 nt, and 9 nt and 8 nt. (e) An example of signatures generated with SSA and added Gaussian noise. (f) A sample temporal barcode showing the on-time (t_{on}) and off-time (t_{off}) in a signature. (g) A scatter plot for ssDNA devices with data collected for 10 and 30 minutes. The blue shade represents each device can be clustered effectively with longer data-collection times.

rate constants can be found in the Methods section. For each temporal barcode collected, the length of their on-time peaks was collected and a histogram which was approximated by an exponential distribution, as shown in Fig. 3d. The on-time peaks represent the amount of time when a fluorescent reporter strand is attached to the DNA device which is also the amount of time system is in the state S1 (also called the binding time) in Fig. 3b. The estimated rate value for each exponential distribution that best represents data for different collection times is shown in Fig. 3c. The error bars indicate an interval for estimating the mean value of the distribution with 95% confidence. The histogram plots in Fig. 3d show on-time distribution fits to an exponential probability density function (pdf) for a collection time of 60 minutes.

The log plot shown in Fig. 3c clearly shows the difference in the estimated rate values for each DNA device. This means that each device can be distinguished by estimating their average on-time parameter. Additionally, the difference in the estimated value of the mean on-time is also significant demonstrating the potential space that can be packed with several new DNA devices by designing a sophisticated set of DNA devices. Also note that the error bars of the estimated parameter gets tighter as the data collection time is increased. This is expected as the summary statistics of memoryless stochastic process can be approximated better as the samples size increase. This can also be achieved by increasing the frequency of hybridization if shorter data collection time is critical for reporting application. Several other physically tunable parameters can, therefore, be exploited to achieve a much finer distribution of the range while ensuring sufficient distinguishability. Fig 3e shows a sample temporal DNA barcode for 8, 9 and 10 nt device with added Gaussian noise for visualization purposes. A quick visual inspection of devices in Fig. 3f with length 9 nt and 10 nt also indicates distinguishable behavior.

For further analysis of data, the same simulation experiment was repeated for 100 samples one for each 7, 8, 9 and 10 nt device with data collection times 10 minutes and 30 minutes. The mean on-times for each temporal barcode was estimated and is shown in Fig. 3g. As

shown in the figure, there is a significant overlap among sample points of different devices if the data collection time is only 10 minutes at the given concentration of fluorescent reporter strand. However, if the data collection time is increased to 30 minutes, the samples are further separated allowing us to easily cluster them using a simple spatial clustering algorithm such as k-mean, nearest neighbor etc.[42]

Note that 10^{-4} on the vertical axis scale is numerical zero for the scatter plots shown in Fig. 3 indicating the detection limit for our simulation experiments. Also note the red dot at 10^{-4} seconds in Fig. 3g. It represents no output on-peaks in the temporal barcode of a device and therefore the estimated on-time for that sample point is numerical zero. Since the process is stochastic, this is possible but very unlikely as seen in only one of the hundred samples recorded. It can be avoided by longer data-collection times as observed in the next figure with collection time 30 minutes.

Tuning the number of domains to create diversity

After tuning the length parameter, we next tune the temporal barcode of a ssDNA with multiple domains as summation of temporal barcodes of the constituent domains for additional programmability as shown in Fig. 4. A simple two domain device will have three observable states and an unobservable state, termed as the dark state, as represented with a 4-state CTMC model in Fig. 4b, where states $S1$ and $S3$ each represents one of the device domains bound to fluorescent strand. The additional state $S2$ represents the device with multiple fluorescent strands hybridized at the same time. Such state will have a visible jump in the fluorescence intensity since the emitted photon count is linearly proportional to the number of fluorescent dyes.[43]

We represent a double domain device using the notation $(|a|, |b|)$ where $\{|a|, |b|\} \in S$ and $|x|$ indicates the length of the domain. Note that we cannot control the order in which reporter strands attach to our devices hence $(|a|, |b|) = (|b|, |a|)$. We performed a simulation experiment with 10 different devices with rates similar to previous section. Like prior simulation

experiments, we restricted our domain lengths from 7 to 10 nt *i.e.* $S = \{x | x \in [7, 8, 9, 10]\}$. We analyzed all the output signals to compute two parameters: (a) the on-time (t_{on}) and (b) the double-blink time (t_{db}). We define double-blink time as the amount of time when both the fluorescent strands are attached to our devices. A histogram plot for both these parameters was constructed and an exponential distribution was fitted to extract the rate parameters (or mean) of these distributions. The entire process was repeated for a few hundred samples and a 2D plot of the estimated parameters for all the simulated devices are shown in Fig. 4c. When the data collection time was 30 minutes, some of the shorter devices had an overlap in the scatter plot. However, this was easily resolved with an increase in the data collection time. A data collection time of roughly 60 minutes at 25 nM fluorescent strand concentration allows us to easily classify these 10 devices with high accuracy. For some shorter devices, there are samples without any double-blinks but these are still separable.

Note that we restricted the simulation experiments to devices 7 nt and longer since most detectors can only capture events longer than 1 ms. However, if a CMOS camera is used, one could easily integrate shorter devices to increase the pool of distinguishable devices.[22] Finally, note that 10^{-3} on the vertical axis scale is numerical zero for the scatter plots shown in Fig. 4. They signify that the barcode signatures did not have any double-blink.

Tuning the order by domain sequestering to create diversity

An interesting functionality of secondary structures such as DNA hairpins is their ability to sequester information. As an improvement over ssDNA devices, this programmability can be useful to enforce a binding order of the reporter strands. This can help differentiate between devices $(|a|, |b|)$ where domain a is exposed, and b is sequestered, and $(|b|, |a|)$ where domain b is exposed and a is sequestered, thereby increasing the number of distinguishable devices. Therefore, by simply reversing the order of reporter domains, we can approximately double the number of devices.

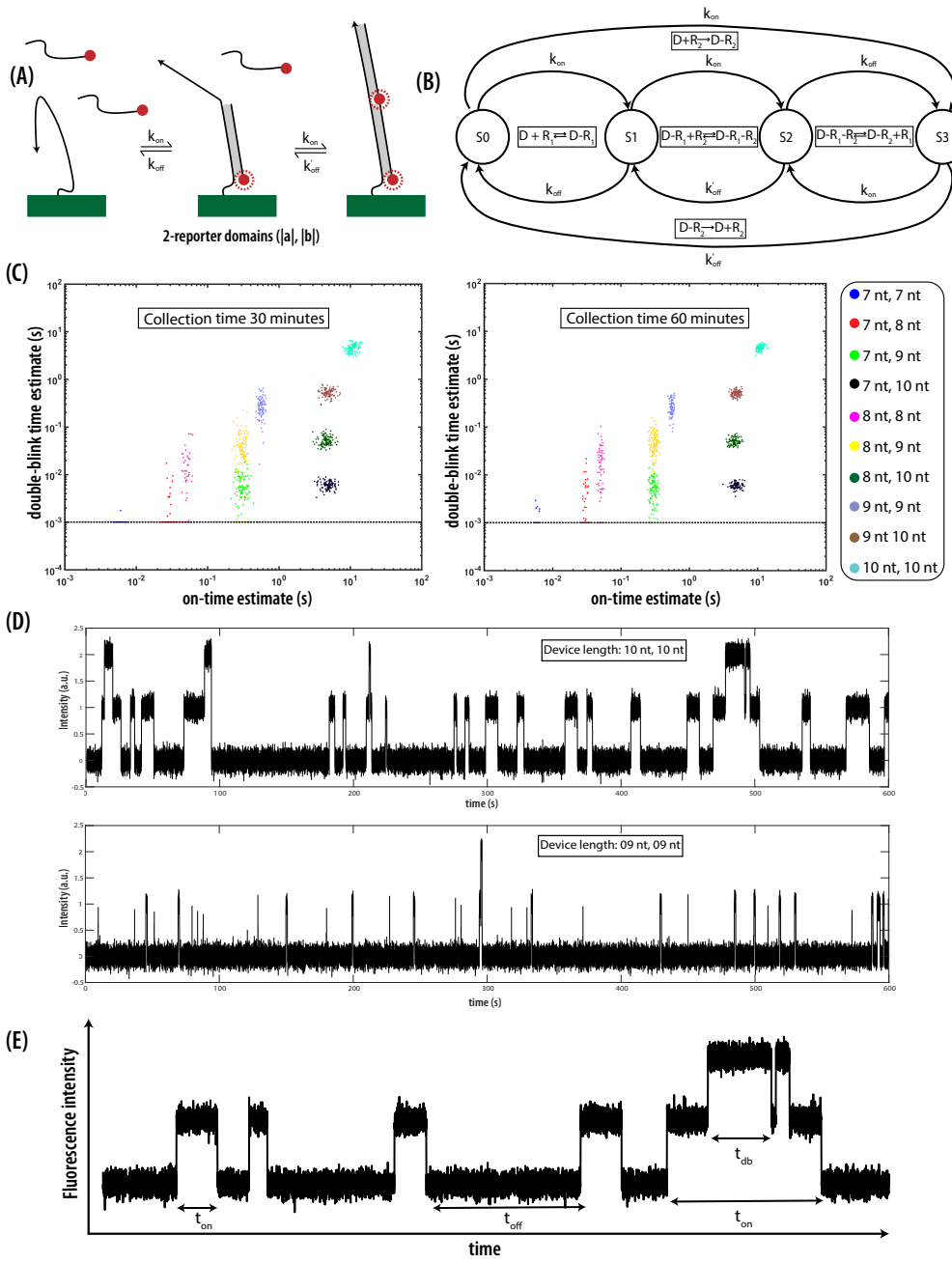

Figure 4: Tuning the number of domains to program temporal barcodes. (a) The transient binding of fluorescent strands to our DNA device. (b) A 4-state CTMC model representation. (c) A scatter plot in the parameter space generated by learning parameters from intensity signatures of 10 different devices. (d) A typical signature of (10,10) and (9,9) device collected for 10 minutes. (e) A sample temporal barcode showing on-time (t_{on}), off-time (t_{off}) and double-blink time (t_{db}).

The model for hairpin-based devices with two domains is very similar to prior ssDNA devices with two domains as shown in Fig. 4b and Fig. 5b. The only difference is a fluorescent

reporter's inability to bind with the hidden domain without successfully opening the hairpin. A simulation experiment with 25 nM fluorescent strand was performed with similar rate parameters as prior sections for individual domains. The noisy output signal was analyzed to compute the following parameters: (a) single-step time (t_{ss}), (b) double-blink time (t_{db}), and (c) double-step time (t_{ds}). A histogram was generated by analyzing all the signals to compute all 3 parameters. The exponential distributions best approximating these histograms produced estimated mean values with 95% confidence.

Note that t_{ss} and t_{ds} are computed differently as shown in Fig. 5d. We compute the single-step time by calculating the on-time for all the peaks that had exactly one reporter strand attached to it while the double-step time here refers to the on-time time for all the peaks with double-blink time greater than zero. A 3D scatter plot in parameter space for all possible device combinations of 7 to 10 nt domain length is shown in Fig. 5. The scatter plot of data collected for 200 minutes can easily be classified using popular clustering algorithms such as k-mean, mean-shift etc.[42] with high accuracy. Note that for some of the shorter devices there are samples where no double-blink was observed. Therefore, devices with at least one longer domain is the preferred choice when designing such DNA devices to report single-molecules. Note that this strategy assumes that we design the hairpin sequence such that after annealing it remains as a stable hairpin. This can be ensured by having longer stems. Additionally, prior studies also suggest longer hairpin stem for higher stability and therefore lower leak.[36]

Tuning the dark-time with a competing secondary structure

It is a well-known phenomenon that ssDNA can also be programmed to form a secondary structure such as the DNA-hairpin if complementary sub-sequences exists.[23] This is helpful since it gives more room for programming signatures of DNA devices. Such competing secondary structure changes the dark-time (t_{off}) of the temporal barcode. As shown in Fig. 6, a DNA device with complementary sub-sequence can form a DNA hairpin which

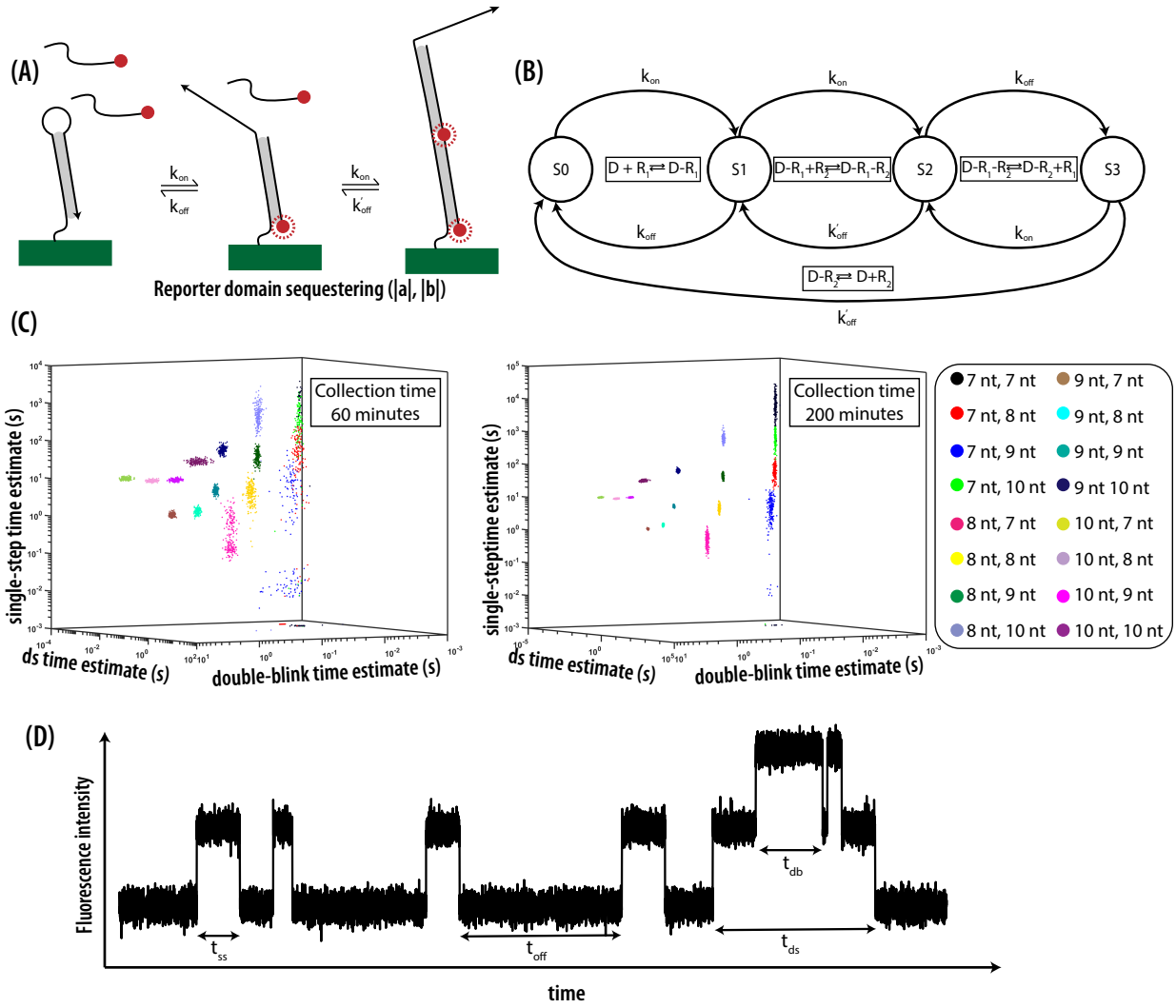

Figure 5: Tuning the sequence of domains to program temporal barcodes. (a) The transient binding of fluorescent strands to our DNA device with hairpin secondary structure such as DNA hairpin to sequester a domain. (b) A 4-state CTMC model representation. (c) A 3D scatter plot in the parameter space generated by learning parameters from intensity signatures of 10 different devices. (d) A typical signature of a device indicating the difference between the calculated parameters single-step time and double-step time.

inhibits attachment of the fluorescent reporter. Therefore, we modeled this system using the 3-state CTMC as shown in Fig. 6b and performed a simulation experiment with rates for hairpin closing adopted from Tsukanov et al.[36] A fluorescent reporter of length 10 nt was allowed to interact with the devices that can form hairpins with a stem length of 6 to 10 nt. The estimated dark-time for all the simulation experiments with a data collection time of 90 minutes yielded a distinguishable device set as shown in Fig. 6c. These type of devices are

extremely important since most existing multiplexing techniques that do not use wavelength multiplexing, encode information in the DNA sequence.[21] Therefore, they need multiple dye-labeled DNA strands which increases the experimental costs significantly. With our technique, only a single dye-labeled DNA strand is required for multiple reporting devices making this reporting technique highly cost-effective.

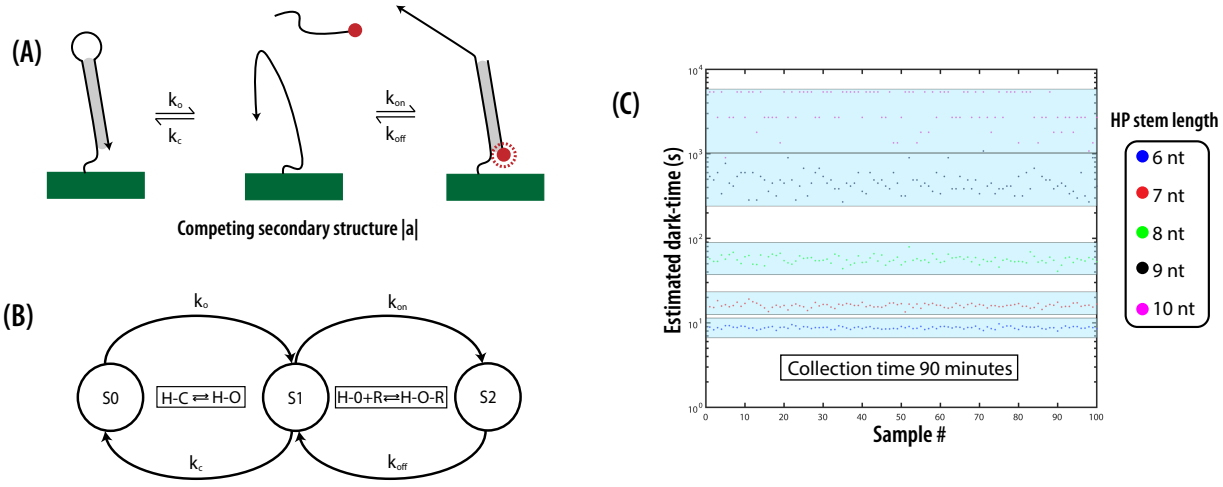

Figure 6: Tuning the secondary structure of the device to program temporal barcodes. (a) The transient binding of fluorescent strands to our DNA device which can inhibit this process if it forms a secondary structure. (b) A 3-state CTMC model representation. (c) A scatter plot in the parameter space (dark-time) generated by learning parameters from intensity signatures of 5 different devices. Length of fluorescent strand was constant at 10 nt while hairpin stem length ranged from 6 to 10 nt.

Enhanced tunability with nanostructures

We have so far only tuned individual DNA devices in this work, however, the tunability of our temporal barcoding framework can be further scaled up by using a nanostructure as a breadboard. The field of structural DNA self-assembly is sophisticated enough to produce structures ranging from a few nanometers to several hundreds of nanometers.[44] Several complex shapes have been proposed and it is straightforward to construct a simple breadboard-like DNA rectangle with length under 200 nm using the DNA origami technique.[23] Some of the staples can be extended to act as DNA devices attached to the breadboard. The nanostructure can act as one DNA device and will have overlapping point spread function

(PSF) for individual devices on its surface as this structure is smaller than the diffraction limit of light. While this property is undesired for the field of super-resolution imaging,[3,4] it can be exploited here to achieve a large set of temporal barcodes using a linear combination of individual device barcodes.

Using a nanostructure, there are mainly two tunable barcode regimes: (a) low reporter concentration, and (b) high reporter concentration. In regime (a), the reporter concentration is set to be sufficiently low so that only one reporter binds at a time. This regime is also widely used by SMLM techniques to localize each spot with high accuracy.[20] In regime (b), the reporter concentration is much higher and, therefore, multiple reporters can bind at the same time. The behavior of each regime can be visualized Fig. 7b. A simple DNA origami rectangle-like nanostructure can be modified to have four extended staple strands, as shown in Fig. 7a, each acting as a DNA device. A sample temporal barcode for each device and their observed accumulated effect in both the regimes described above is shown in Fig. 7b. As shown in Fig. 7a and Fig. 7b, each modified staple acts as a DNA device for a reporter to attach. However, since all the staples are within the diffraction limited zone and a reporter with only single-dye is present, the output temporal barcode will be a linear combination of individual temporal DNA barcodes.

A simple simulation experiment with a small set of devices was run to study the effect of nanostructure-based temporal barcodes. Assuming the nanostructure shown in Fig. 7a, several output device combinations are possible. Here we take a small subset of the device space (about 12 device combinations) to generate temporal barcodes in the low-reporter regime and high-reporter regime. More details about the simulation experiment parameters can be found in the Methods section. The output temporal barcodes for each regime were analyzed to extract on-time and double-blink parameters like prior simulation experiments. The scatter plots with estimated parameters are shown in Fig. 7c and Fig. 7d demonstrating the distinct behavior of the temporal barcodes set constructed using a linear combination of individual devices. Note that we only used partial information, on-time for level one and two,

for the temporal barcode in the high reporter regime for visualization purposes. Additionally, we manually chose a small subset of device combinations for demonstration purposes, however, the problem of choosing a set of non-overlapping devices can be reduced to the classic NP-complete maximal set packing problem and can be solved by greedy algorithms.[45]

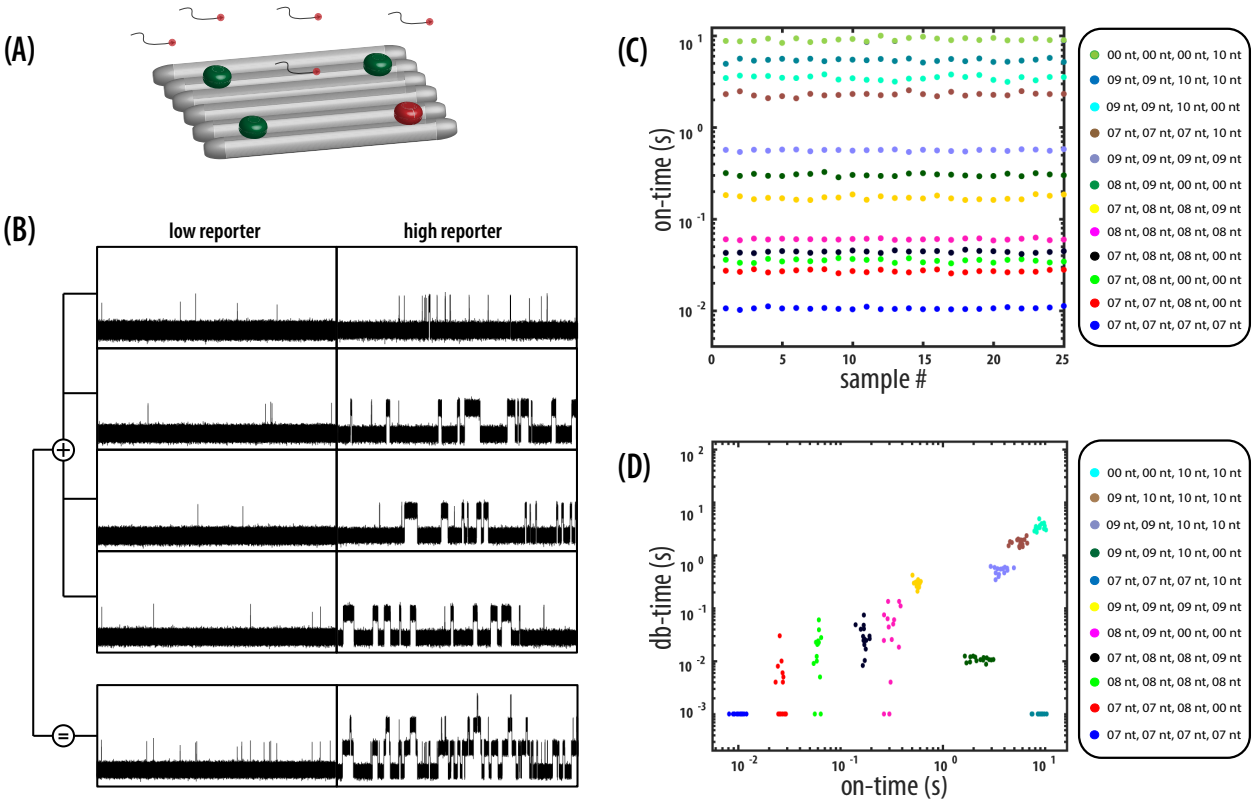

Figure 7: Using a nanostructure as a breadboard to design a set of temporal DNA barcodes. (a) An illustration of a DNA nanostructure with 4 staple modifications each of which can act as a DNA device. The red spot on the nanostructure indicates a fluorescent reporter bound to the device. (b) A sample temporal barcode in the low and high reporter regime generated using four devices on a nanostructure. Note the difference in the final temporal DNA barcode of the nanostructure in each regime. (c) A small subset of the possible device combinations was chosen to generate 12 different temporal DNA barcodes using the low reporter regime. (d) A small subset of the possible device combinations was chosen to generate 11 different temporal DNA barcodes using the high reporter regime.

Using supervised machine learning for barcode classification

We demonstrate the scalability of the temporal barcoding framework by running a concrete large-scale experiment. Using the design principles of our framework, we ran a simulation ex-

periment which assumes a DNA nanostructure such as DNA origami rectangle with five staple modifications. Each of these staples can be one of the simple devices $\{7nt, 8nt, 9nt, 10nt\}$ and they all attach to the same universal complementary reporter strands floating in the solution. The DNA origami rectangle design[43] is about 90 nm tall and 70 nm wide and, therefore, the temporal barcode observed will be a linear combination of individual device barcodes as this is smaller than the diffraction limit of light. The reporter concentration for our simulation experiment was kept high at 30 nM so that our temporal barcodes would operate in the high reporter regime, as shown in Fig. 7b. The total number of unique combinations of devices is given by $\binom{N+K-1}{K}$, where K is the number of staples and N is the number of simple DNA devices.[46] This is result of the counting the number of ways of to distribute K objects into N distinct cells such that $c_1 + c_2 + \ldots + c_N = K$. For example, if we have 5 staple extensions and 4 different devices, the total number of unique combinations will be $\binom{4+5-1}{5} = 56$ and up to five observable on-states with quantized intensities.

For successful identification of each device, we extract multiple on-times from their temporal barcode, with one on-time per level as shown in Fig. 8a. For a temporal barcode with k+1 levels, there will be k bright states or on-states and one dark state. In our experiment, as each nanostructure has five devices, each temporal barcode will have up to five on-states and one off-state. We can measure the average on-time for each bright state and construct a five-dimensional input feature vector for each temporal barcode. Each value in the feature vector stands for the estimated on-time obtained by fitting an exponential distribution to the histogram plot. This vector can be fed as an input to a machine learning model for automatizing the classification (or barcode to device identification) process. In this work, we will train an SVM model with Gaussian kernel for the classification of temporal DNA barcodes. A feature vector of a temporal barcode is a $[\mathrm{a}_i]_{k\times 1}$ matrix where the i^{th} row indicates the average on-time of i^{th} level in the temporal barcode. This $k+1$ state temporal barcode will have a $k \times 1$ feature vector as we ignore the dark state. We ran a 5-hour simulation experiment for each nanostructure to collect the corresponding temporal DNA barcode and

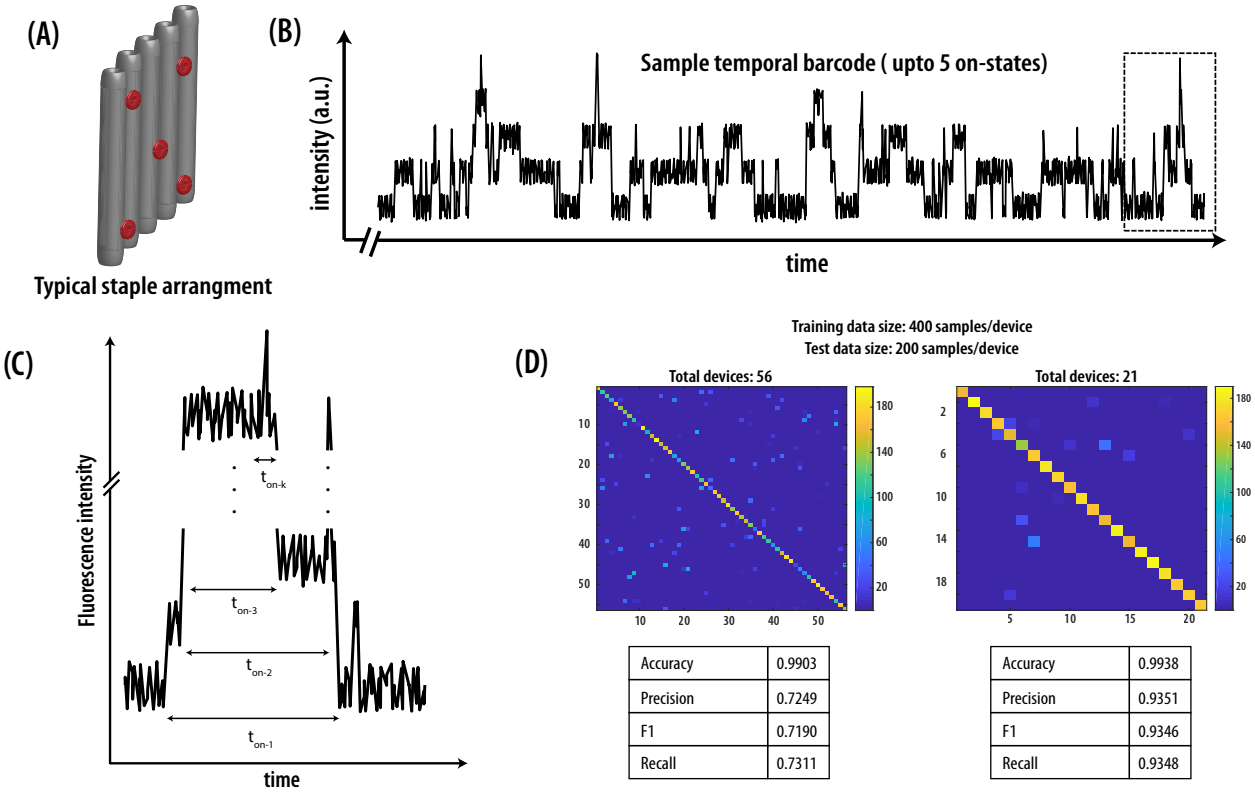

Figure 8: Training a non-linear SVM classifier on a large pool of temporal DNA barcodes. (a) A typical nanostructure with five extended staples. The location of staples can be modified as long as the inter-staple distance is > 10 nm. (b) A typical temporal barcode with up to 5 states. (c) In general, a nanostructure can have k devices and can generate a temporal barcode with up to k-levels. The corresponding on-time measured for each level will form a k-dimensional feature vector for a classification model. (d) A confusion matrix plot generated using the test data set on a trained SVM model. All the important evaluation parameters - accuracy, precision, F1 score and recall - are summarized in the tables for each confusion matrix plot.

analyzed each barcode to generate a feature vector. Using MATLAB's deep learning toolbox, we trained our SVM model and evaluated its performance on the test data set. More details on simulations and scripts can be found in the Methods section. The confusion matrix for the test data set is shown in Fig. 8b along with the evaluation scores, namely, accuracy, precision, F1 and recall. Most of the weight lies on the diagonal of the matrix indicating good overall classification performance and close to perfect accuracy. However, there are some misclassifications giving us a little over 70% precision. This is mainly due to overlap in shorter devices as their events are short-lived and unlikely. Based on the results of simple de-

vices shown in prior sections, we already saw that to achieve higher distinguishability, longer devices are preferred in the high reporter regime. This is mainly because shorter devices lead to short-lived events and therefore unlikely higher-dimensional values in the feature vector space. Therefore, we mask out all the device combinations that contain 7 nt devices and re-ran the model training and evaluation process with the subset of combinations. Since we have $N = 3$ type of devices and $K = 5$ staple buckets, the total number of combinations will be $\binom{3+5-1}{5} = \binom{7}{2} = 21$. The confusion matrix for these devices is shown in Fig. 8b. As expected, we achieve a much higher classification performance with over 90% F1-score. This shows the ability of the trained model to distinguish our designed device set.

Discussion

In this work, we have introduced a novel time-based framework for designing a family of DNA-based devices, for unique identification of the single-molecule. These devices undergo a series of dynamic transformations that result in a unique temporally-varying fluorescence signal. Since they encode information in the time domain, we can design several devices with as few as one-dye greatly simplifying the hardware setup for data collection. These devices are easy to design and require only one universal fluorescence reporter strand making them extremely cost-effective. In addition, they follow the principle of transient binding which makes them relatively immune to photo-bleaching when imaged using TIRF microscopes.

Our framework introduced five different design methodologies to generate several distinguishable temporal barcodes, namely (a) tuning the device length (b) tuning the number of domains (c) tuning the order by domain sequestering (d) tuning the dark-time with competing secondary structure formation and (e) using nanostructures for linear combination. Each of these design principles was then used to generate a family of DNA devices with different barcodes using only one fluorescent dye. We modeled the behavior of our DNA devices using CTMCs and performed several simulation experiments to demonstrate our idea

and identify experimental conditions for maximal distinguishability. Finally, using simple principles of our framework, we showed an end-to-end large-scale nanostructure example by training an SVM model for non-linear classification, indicating the robustness of our framework. Nearly 100 temporal DNA barcodes were designed, modeled, simulated and analyzed in this work. Although our barcodes can work with as few as one dye, by adding multiple dyes, we can create much larger families of uniquely identifiable reporter molecules which makes our framework highly scalable.

Scaling the number of unique barcodes

Although our simulation experiments were made using only one type of dye, here we estimate the number of unique barcodes we can make with the use of multiple dyes to demonstrate the robustness of our technique. In the previous subsection, we already showed the number of temporal barcode combinations for a nanostructure with K staple extensions, for N tunable device lengths, is given by $\binom{N+K-1}{K}$. However, if we use multiple dye colors (for example, D), the total number of unique temporal barcodes further scales up. A realistic value for D can be 4 with a sample dye set containing ATTO 405, ATTO 488, Cy3B, ATTO 655 giving us a total of $\binom{D\times N+K-1}{K}$ devices. For the suggested values of $D = 4$, a simple nanostructure with 5 staple extensions can generate $\binom{4\times 4+5-1}{5} = \binom{20}{5} = 15504$ combinations. While this design space certainly has overlapping barcodes, it should be noted that it only tunes the length of ssDNA devices residing on a DNA origami-like platform. We can also use a prior technique, that utilizes the geometry of nanostructures[7] in combination with our temporal encoding to scale this number even further.

Error-correcting optical barcodes using DNA nanostructure

A natural application of our temporal barcoding framework is DNA nanostructure tagging as shown in earlier sections. With nanostructures such as a 6-helix bundle,[7] error-correcting optical barcodes can also be created as this structure is longer than light's diffraction limit.

As shown in Fig. 9, the ends of a 6-helix bundle can be tagged with two devices (of similar or different types), which can independently report the tagged structure. If error correction is desirable in the detection application; each nanostructure can be tagged with the same device multiple times because the identification of even a single temporal barcode should uniquely identify the structure of interest. Such redundant multi-tagging can also ensure that the reporting occurs even if an origami nanostructure has incomplete binding fidelity of all staples.

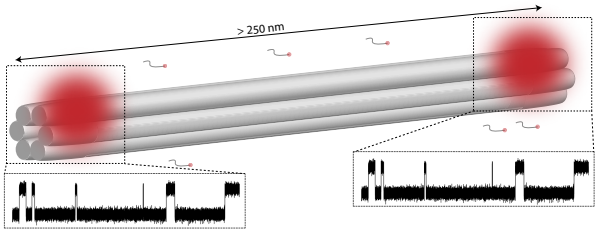

Figure 9: A 6-helix bundle can be tagged with our temporal reporters (without using wavelength multiplexing) in two regions to generate a unique temporal signature. These types of nanostructure tagging can also be error-resistant since only one of the two devices is required.

Insights for experimental demonstration

Our simulation experiments suggest a few key points which can help expedite the experimental demonstration: (a) Longer domain length is preferred. This will enable relatively slower dynamics of the molecular process enabling a higher signal-to-noise ratio for the signal detected. (b) Fluorescence microscope in TIRF mode is preferred. This will reduce the background fluorescence arising from the free-floating reporter strands. (c) Longer data acquisition is preferred. This will help to reduce the variance of the acquired temporal barcodes and eventually the estimated statistical parameters. (d) Extremely low device concentration is preferred. This is crucial as our devices are much smaller than the diffraction limit of light. Since the device location of surface binding is random, it is possible that multiple devices sit in the vicinity of each other. This can be avoided by reducing the concentration of DNA devices as it can make the likelihood probability extremely small. (e) False positive should be discarded. Since the overall goal of the study is the development of taggants, accurate identification of the DNA device is crucial. Therefore, a strict software or hardware protocol is essential to discard all the non-specific fluorescent events recorded.

The decision of which reporter regime to work with depends on the available experimental setup and the target number of devices in the set. This is because the high variance in the estimated parameters due to shorter devices can lead to poor model performance if we are working in the high reporter regime as this regime prefers cascading events. If the devices are shorter, they are not conducive to generating higher order events since their binding events are short-lived. However, such short events are desirable in the low-reporter regime since the regime aims to avoid cascaded events completely.

Cost-effectiveness of temporal DNA barcodes

A hypothetical experiment design can demonstrate the cost-effectiveness of our barcoding technique. For example, if the experiment requires ten-fold multiplexing, we need to design a pool of ten orthogonal temporal DNA barcodes. This requires ten DNA strands (approx. \$10 each, $<$ \$0.1 per bp) and one dye-labeled universal reporter (approx. \$250) bringing the total cost to \$250 + \$100 = \$350. In comparison, an Exchange-PAINT experiment with ten-fold multiplexing will require ten reporters and therefore the approximate cost will be \$2500 + \$100 = \$2600 demonstrating the cost-effectiveness of our technique.

Methods

Generating temporal barcode for individual devices

All the simulation experiments were conducted using custom-written MATLAB scripts available as part of the supplemental material. Briefly, we used MATLAB's SimBiology toolbox to represent our Markov models as chemical reactions and simulated them using SSA algorithm also available as a part of the SimBiology package. The programming details can be found in the supplementary scripts and MATLAB's online documentation. For further resemblance to experimental data, we added Gaussian noise to the simulated state chain since the combined effect of shot noise, dark noise and all other detector noises are usually

approximated by a Gaussian distribution.[47] The unbinding rate constants for 7 nt, 8 nt, 9 nt and 10 nt were 10 ms, 60 ms, 550 ms, and 9 s respectively. The binding rate constants for all the simulation were 10^6 $M^{-1}s^{-1}$. More details on rate constants can be obtained from prior literature.[21,22,37] The simulated state chain output was analyzed to compute parameters such as on-time, off-time, double-blink etc. Once all the samples of a parameter were acquired from the state chain, a histogram was constructed and fit to an exponential distribution using maximum likelihood estimator. This can estimate the rate parameter (or mean) of the exponential distribution. Since the overall DNA hybridization process is stochastic, the simulation experiments were repeated several times to generate the scatter plots shown in Fig. 3, Fig. 4, Fig. 5 and Fig. 6.

Generating temporal barcode for nanostructures

To emulate the behavior of a nanostructure, the simulation script independently simulated each device on its surface similar to the prior section and linearly combined their temporal barcodes to resemble experimental behaviors of single-molecules observed under the diffraction limit of light. This kept the simulations simple and modular as the modules used for generating individual device barcode were reused. For low reporter concentration regime, the simulation experiment was kept 8 hours and the reporter concentration set to 1 nM. For high reporter concentration regime, the simulation experiment was kept 5 hours and the reporter concentration set to 30 nM. The simulation experiment was repeated multiple times to generate enough samples for the scatter plot shown in Fig. 7c and Fig. 7d.

Training SVM model for supervised learning

For the nanostructure example, we ran the simulation experiments for each device 600 times giving us a total of 33600 samples for all the combinations of the devices. This ensured that enough data was available for model training and testing purposes. From each sample, a 5-dimensional feature vector was extracted where each dimension indicates the average on-

time for that level. After random shuffling, 70% data samples were fed to the SVM model for training purposes and the remaining 30% of the data samples were left for the testing purposes. Using the gradient descent method, the model was trained to minimize the loss function of the SVM model using the OneVsAll encoding scheme. More details about training an SVM model and model parameters can be found in the online MATLAB documentation for the method fitcecoc. Once the model was trained, we evaluated its performance by generating a confusion matrix plot and calculating several metrics such as accuracy and precision. The open source package confusionmatStats was used to calculate these metrics and MATLAB's built-in method was used to generate the plot.

The cluster machine used to run the simulation experiments and model training had the following configuration: 10x Tensor TXR231-1000R D126 Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.40GHz (512GB RAM - 40 cores).

Acknowledgement

The authors would like to thank Daniel Fu, Xin Song, Ming Yang, Tianqi Song, Abeer Eshra and Hieu Bui for their comments and suggestions. The authors would also like to thank Xin Song for proof-reading the manuscript.

This work was supported by National Science Foundation Grants CCF-1813805 and CCF-1617791.

Author contributions

S.S conducted the study, performed experiments and wrote the paper. A.D analyzed the results and wrote the paper. J.R supervised the study.

Supporting Information Available

The scripts used for modeling, simulation and analysis works are available as part of the supporting information.

References

(1) Sauer, M.; Soeller, C.; Hanrahan, O. Science **2015**, 350, 699–699.

(2) Moore, R. P.; Legant, W. R. Nature methods **2018**, 15, 659.

(3) Huang, B.; Wang, W.; Bates, M.; Zhuang, X. Science **2008**, 319, 810–813.

(4) Sharonov, A.; Hochstrasser, R. M. Proceedings of the National Academy of Sciences **2006**, 103, 18911–18916.

(5) Legant, W. R.; Shao, L.; Grimm, J. B.; Brown, T. A.; Milkie, D. E.; Avants, B. B.; Lavis, L. D.; Betzig, E. Nature methods **2016**, 13, 359.

(6) Bates, M.; Huang, B.; Dempsey, G. T.; Zhuang, X. Science **2007**, 317, 1749–1753.

(7) Lin, C.; Jungmann, R.; Leifer, A. M.; Li, C.; Levner, D.; Church, G. M.; Shih, W. M.; Yin, P. Nature chemistry **2012**, 4, 832.

(8) Nicewarner-Pena, S. R.; Freeman, R. G.; Reiss, B. D.; He, L.; Peña, D. J.; Walton, I. D.; Cromer, R.; Keating, C. D.; Natan, M. J. Science **2001**, 294, 137–141.

(9) Gudiksen, M. S.; Lauhon, L. J.; Wang, J.; Smith, D. C.; Lieber, C. M. Nature **2002**, 415, 617.

(10) Levsky, J. M.; Shenoy, S. M.; Pezo, R. C.; Singer, R. H. Science **2002**, 297, 836–840.

(11) Braeckmans, K.; De Smedt, S. C.; Roelant, C.; Leblans, M.; Pauwels, R.; Demeester, J. Nature materials **2003**, 2, 169.

(12) Dejneka, M. J.; Streltsov, A.; Pal, S.; Frutos, A. G.; Powell, C. L.; Yost, K.; Yuen, P. K.; Müller, U.; Lahiri, J. Proceedings of the National Academy of Sciences **2003**, 100, 389–393.

(13) Li, Y.; Cu, Y. T. H.; Luo, D. Nature biotechnology **2005**, 23, 885.

(14) Lin, C.; Liu, Y.; Yan, H. Nano letters **2007**, 7, 507–512.

(15) Pregibon, D. C.; Toner, M.; Doyle, P. S. Science **2007**, 315, 1393–1396.

(16) Geiss, G. K.; Bumgarner, R. E.; Birditt, B.; Dahl, T.; Dowidar, N.; Dunaway, D. L.; Fell, H. P.; Ferree, S.; George, R. D.; Grogan, T. Nature biotechnology **2008**, 26, 317.

(17) Lu, Y. et al. Nature Photonics **2014**, 8, 32.

(18) Zhang, Y.; Zhang, L.; Deng, R.; Tian, J.; Zong, Y.; Jin, D.; Liu, X. Journal of the American Chemical Society **2014**, 136, 4893–4896.

(19) Shang, L.; Fu, F.; Cheng, Y.; Wang, H.; Liu, Y.; Zhao, Y.; Gu, Z. Journal of the American Chemical Society **2015**, 137, 15533–15539.

(20) Woehrstein, J. B.; Strauss, M. T.; Ong, L. L.; Wei, B.; Zhang, D. Y.; Jungmann, R.; Yin, P. Science advances **2017**, 3, e1602128.

(21) Jungmann, R.; Avendaño, M. S.; Woehrstein, J. B.; Dai, M.; Shih, W. M.; Yin, P. Nature methods **2014**, 11, 313.

(22) Schnitzbauer, J.; Strauss, M. T.; Schlichthaerle, T.; Schueder, F.; Jungmann, R. Nature protocols **2017**, 12, 1198.

(23) Bui, H.; Shah, S.; Mokhtar, R.; Song, T.; Garg, S.; Reif, J. ACS nano **2018**, 12, 1146–1155.

(24) Johnson-Buck, A.; Shih, W. M. Nano letters **2017**, 17, 7940–7944.

(25) Wang, S.; Vyas, R.; Dwyer, C. Optics express **2016**, 24, 15528–15545.

(26) Song, X.; Eshra, A.; Dwyer, C.; Reif, J. RSC Advances **2017**, 7, 28130–28144.

(27) Shah, S.; Limbachiya, D.; Gupta, M. K. arXiv preprint arXiv:1310.6992 **2013**,

(28) Shah, S.; Dave, P.; Gupta, M. K. arXiv preprint arXiv:1502.05552 **2015**,

(29) Stewart, K.; Chen, Y.-J.; Ward, D.; Liu, X.; Seelig, G.; Strauss, K.; Ceze, L. A content-addressable DNA database with learned sequence encodings. International Conference on DNA Computing and Molecular Programming. 2018; pp 55–70.

(30) Chen, Y.-J.; Dalchau, N.; Srinivas, N.; Phillips, A.; Cardelli, L.; Soloveichik, D.; Seelig, G. Nature nanotechnology **2013**, 8, 755.

(31) Cherry, K. M.; Qian, L. Nature **2018**, 559, 370.

(32) Chatterjee, G.; Dalchau, N.; Muscat, R. A.; Phillips, A.; Seelig, G. Nature nanotechnology **2017**, 12, 920.

(33) Garg, S.; Shah, S.; Bui, H.; Song, T.; Mokhtar, R.; Reif, J. Small **2018**, 14, 1801470.

(34) Kopperger, E.; List, J.; Madhira, S.; Rothfischer, F.; Lamb, D. C.; Simmel, F. C. Science **2018**, 359, 296–301.

(35) Fu, D.; Shah, S.; Song, T.; Reif, J. Synthetic Biology; Springer, 2018; pp 411–417.

(36) Tsukanov, R.; Tomov, T. E.; Masoud, R.; Drory, H.; Plavner, N.; Liber, M.; Nir, E. The Journal of Physical Chemistry B **2013**, 117, 11932–11942.

(37) Jungmann, R.; Steinhauer, C.; Scheible, M.; Kuzyk, A.; Tinnefeld, P.; Simmel, F. C. Nano letters **2010**, 10, 4756–4761.

(38) Zadeh, J. N.; Steenberg, C. D.; Bois, J. S.; Wolfe, B. R.; Pierce, M. B.; Khan, A. R.; Dirks, R. M.; Pierce, N. A. Journal of computational chemistry **2011**, 32, 170–173.

(39) Gillespie, D. T. Annu. Rev. Phys. Chem. **2007**, 58, 35–55.

(40) Trivedi, K. S. Probability & statistics with reliability, queuing and computer science applications; John Wiley & Sons, 2008.

(41) Morrison, L. E.; Stols, L. M. Biochemistry **1993**, 32, 3095–3104.

(42) Joshi, A.; Kaur, R. International Journal of Advanced Research in Computer Science and Software Engineering **2013**, 3.

(43) Schmied, J. J.; Raab, M.; Forthmann, C.; Pibiri, E.; Wünsch, B.; Dammeyer, T.; Tinnefeld, P. Nature protocols **2014**, 9, 1367.

(44) Tikhomirov, G.; Petersen, P.; Qian, L. Nature **2017**, 552, 67.

(45) Cormen, T. H.; Leiserson, C. E.; Rivest, R. L.; Stein, C. Introduction to algorithms; 2009.

(46) Guichard, D. An Introduction to Combinatorics and Graph Theory; Whitman College-Creative Commons, 2017.

(47) McKinney, S. A.; Joo, C.; Ha, T. Biophysical journal **2006**, 91, 1941–1951.