A Molecular-scale Programmable Stochastic Process Based On Resonance Energy Transfer Networks: Modeling And Applications

by

Siyang Wang

Department of Electrical and Computer Engineering
Duke University

Date:_____

Approved:

_____
[Alvin R. Lebeck], Supervisor

_____
[Hisham Z. Massoud]

_____
[Loren W. Nolte]

_____
[Galen Reeves]

Dissertation submitted in partial fulfillment of
the requirements for the degree of Doctor
of Philosophy in the Department of
Electrical and Computer Engineering in the Graduate School of
Duke University

2016

ABSTRACT

A Molecular-scale Programmable Stochastic Process Based On Resonance Energy Transfer Networks: Modeling And Applications

by

Siyang Wang

Department of Electrical and Computer Engineering
Duke University

Date:_____

Approved:

_____
[Alvin R. Lebeck], Supervisor

_____
[Hisham Z. Massoud]

_____
[Loren W. Nolte]

_____
[Galen Reeves]

An abstract of a dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Electrical and Computer Engineering in the Graduate School of Duke University

2016

Abstract

While molecular and cellular processes are often modeled as stochastic processes, such as Brownian motion, chemical reaction networks and gene regulatory networks, there are few attempts to program a molecular-scale process to physically implement stochastic processes. DNA has been used as a substrate for programming molecular interactions, but its applications are restricted to deterministic functions and unfavorable properties such as slow processing, thermal annealing, aqueous solvents and difficult readout limit them to proof-of-concept purposes. To date, whether there exists a molecular process that can be programmed to implement stochastic processes for practical applications remains unknown.

In this dissertation, a fully specified Resonance Energy Transfer (RET) network between chromophores is accurately fabricated via DNA self-assembly, and the exciton dynamics in the RET network physically implement a stochastic process, specifically a continuous-time Markov chain (CTMC), which has a direct mapping to the physical geometry of the chromophore network. Excited by a light source, a RET network generates random samples in the temporal domain in the form of fluorescence photons which can be detected by a photon detector. The intrinsic sampling distribution of a RET network is derived as a phase-type distribution configured by its CTMC model. The conclusion is that the exciton dynamics in a RET network implement a general and important class of stochastic processes that can be directly and accurately programmed

and used for practical applications of photonics and optoelectronics. Different approaches to using RET networks exist with vast potential applications. As an entropy source that can directly generate samples from virtually arbitrary distributions, RET networks can benefit applications that rely on generating random samples such as 1) fluorescent taggants and 2) stochastic computing.

By using RET networks between chromophores to implement fluorescent taggants with temporally coded signatures, the taggant design is not constrained by resolvable dyes and has a significantly larger coding capacity than spectrally or lifetime coded fluorescent taggants. Meanwhile, the taggant detection process becomes highly efficient, and the Maximum Likelihood Estimation (MLE) based taggant identification guarantees high accuracy even with only a few hundred detected photons.

Meanwhile, RET-based sampling units (RSU) can be constructed to accelerate probabilistic algorithms for wide applications in machine learning and data analytics. Because probabilistic algorithms often rely on iteratively sampling from parameterized distributions, they can be inefficient in practice on the deterministic hardware traditional computers use, especially for high-dimensional and complex problems. As an efficient universal sampling unit, the proposed RSU can be integrated into a processor / GPU as specialized functional units or organized as a discrete accelerator to bring substantial speedups and power savings.

Contents

List of Tables

List of Figures

Acknowledgements

Many people have gave me support during my Ph.D study and contributed to the completion of this thesis. I would like to express my sincere gratitude to them.

First of all, I am grateful to my parents for their love and encouragement. They gave me the freedom and strength to pursue my true passion in life.

I would like to express my deepest gratitude to my advisor Prof. Chris Dwyer for his valuable guidance and support during my study at Duke. His broad knowledge and enlightening insights and ideas have always fascinated and inspired me, and taught me to appreciate the infinite possibilities of science. I am fortunate to have joined Duke Self-Assembled Systems Group and worked with him.

I also want to thank Prof. Alvin Lebeck for this valuable advice on the aspect of computer architecture. It has been a wonderful experience to work with him and everyone in our stochastic computing group. I am very grateful to him for his advice and help with the writing of this thesis and my defense. I would also like to thank Prof. Hisham Massoud, Prof. Loren Nolte and Prof. Galen Reeves for taking time to serve on my committee and providing helpful feedback.

Last but not least, I would like thank all my friends and colleagues, especially my labmates at Duke Self-Assembled Systems Group. We have had so many inspiring discussions. They made my graduate study at Duke an enjoyable experience.

1. Introduction

Stochastic process is often used as a mathematical language to describe molecular and cellular processes, based on which their dynamics and behaviors are studied through theoretical analysis and simulation. However, the inverse problem of programming a molecular-scale process to physically implement general stochastic processes and its potential applications remain mostly unexplored.

This dissertation uses the exciton dynamics in a Resonance Energy Transfer (RET) network to physically implement continuous-time Markov chains (CTMCs), a general and important class of stochastic processes. The implemented stochastic process can be directly programmed through the physical geometry of a RET network which can be accurately fabricated via DNA self-assembly. Different methods of using the RET based stochastic process exist with vast potential applications. Excited by a light source, a RET network generates random samples in the temporal domain in the form of fluorescence photons which can be detected by a photon detector, and the intrinsic sampling distribution of a RET network is a phase-type distribution configured by its CTMC. Based on these observations, this dissertation focuses on using RET networks as an entropy source to directly generate random samples from virtually arbitrary distributions. As illustrated in two real world applications: 1) fluorescent taggants and 2) stochastic computing, the RET-based programmable entropy source can facilitate and

improve applications that rely on generating random samples to perform functions such as pattern recognition and probabilistic algorithms.

1.1 Stochastic models of molecular and cellular processes

Molecular and cellular processes are commonly described as stochastic processes to study and explain their dynamics and behaviors in physics, chemistry and biology. The first extensively investigated molecular-scale process is Brownian motion which was first observed by Robert Brown in 1827 when studying the constant jittery motion of pollen grains within water [1]. After Einstein explained from a physics perspective that the particles were constantly bombarded by water molecules [2], Norbert Wiener rigorously mathematically modeled Brownian motion as the Weiner process, an important continuous-time stochastic process [3].

Besides Brownian motion, stochastic processes are commonly used to model and study chemical reactions at the molecular scale. A chemical system involving multiple coupled chemical reactions can be described as a Chemical Reaction Network. The kinetics of a CRN are often analyzed on a macroscopic level with the law of mass action using deterministic chemical rate equations in the form of ordinary differential equations (ODEs) [4-6]. However, this approach is based on the assumption that the concentration of all reactants varies both continuously and differentiably, and fails when the number of reactant molecules is small. To correctly capture the intrinsic stochastic fluctuation of low-copy-number chemical reactions, their dynamics should be described

as counting processes by Chemical Master Equations (CMEs) and modeled as a stochastic process, specifically a continuous-time Markov chain (CTMC) where the state is a vector comprising of the molecular count of each chemical species and the transition corresponds to a possible reaction [7-10]. The stochastic process often needs to be analyzed to derive the time-dependent probability distributions of different species or infer unknown parameters (e.g., reaction rates) from experimental data. While the state space of the chemical reactions CTMC is usually large and renders the exact solution infeasible, a multitude of stochastic and numerical simulation approaches have been investigated to provide an approximate solution [11-13].

While the CTMC is often a mathematically valid model for CRNs when their reactions follow a Poisson process, it usually becomes incompatible when studying gene regulatory networks (GRNs) and cellular processes. A GRN includes a collection of molecular regulators (e.g., DNA, RNA) and describes their interactions with each other or other substances in the cell to control the gene expression levels of mRNA and proteins. Time delay is a critical aspect for biological reactions such as gene transcription and translation, and a stochastic process with such delays is non-Markovian and requires special treatment for simulation [14-16].

1.2 Molecular-scale implementation of stochastic processes

Although stochastic processes can often theoretically model molecular-scale processes for analyzing their dynamics and inferring parameters, the inverse problem of

physically implementing stochastic processes using a molecular-scale process and its applications remain mostly unexplored. This dissertation explores the exciton dynamics in a Resonance Energy Transfer (RET) network between chromophores as a molecular-scale process that is 1) capable of implementing a general class of stochastic processes, 2) easily programmable and 3) suitable for practical applications.

A chromophore is a single-molecule optical device that can be excited by absorbing photons of a specific wavelength and de-excite through the radiative pathway (i.e., fluorescence at a longer wavelength) and nonradiative pathways, and the exciton relaxation follows temporal exponential distributions. When two chromophores are placed a few nanometers apart and their emission and excitation spectra overlap, energy transfer can occur between the two chromophores through RET. RET is a quantum mechanical energy-transfer mechanism between two chromophores, in which the donor chromophore, initially in its excited state, transfers its energy to the acceptor chromophore through nonradiative dipole-dipole coupling [17]. The time to RET transfer, after the donor is excited, follows an exponential distribution between a chromophore pair, and the transfer rate depends on the characteristics of the chromophores and their distance among other parameters as specified in the Förster equation.

The exciton dynamics in a network of chromophores comprise of the sequence of RET transfers and the sojourn time of each RET transfer, which become a stochastic

process, specifically a continuous-time Markov chain [18]. An important and unique feature of the CTMC is the direct mapping between the physical geometry of a chromophore network and the transition matrix of its corresponding CTMC. The state space of the CTMC is composed of transient states and absorbing states. Each transient state corresponds to a specific chromophore being excited, and the transition rate between a pair of transient states is the RET transfer rate between the corresponding chromophore pair. Each absorbing state corresponds to the exciton leaving the RET network through a specific relaxation pathway, and the transition rate between a transient state and an absorbing state is the decay rate of the corresponding relaxation pathway of the corresponding chromophore.

Based on the above observations, RET networks can physically implement general CTMCs in an explicit and intuitive way that is unprecedented for a molecular-scale process. CTMCs are a general and important class of stochastic processes that has a vast literature of theory and modeling applications in broad fields such as computer networks and distributed systems, chemical reactions, economics and epidemiology, and produce phase-type distributions which can approximate virtually arbitrary distributions. Although chemical reactions at the molecular level could also correspond to a CTMC, the implementation based on exciton dynamics takes a direct approach that is not constrained by the CRN description of a reaction process, and the CTMC implemented by the molecular-scale process can be fully specified in terms of its state

space and transition matrix. With clear physical interpretation, the state space of a RET network implemented CTMC is proportional in size to the chromophore network, and the transition rate between each pair of states is known from the rates of RET transfer and exciton relaxation.

DNA self-assembly and the direct mapping between a RET network and its CTMC facilitate the programmability of the molecular-scale stochastic process. Based on this mapping, the physical specifications of a RET network such as network size, chromophore types and the distance between each chromophore pair become available parameters for programming the CTMC. After fully specifying its physical geometry, an ensemble of the RET network can be accurately and conveniently fabricated via DNA self-assembly. This is the first realization of accurately programming a molecular-scale CTMC with subnanometer precision.

Meanwhile, the RET network implemented CTMC is easily compatible with practical applications of photonics and optoelectronics as a molecular photonic device. Because exciton dynamics between chromophores are composed of RET transfer and fluorescence that often occur on the nanosecond timescale, the physical stochastic process is fast enough for most applications including those that emphasize performance. Because exciton dynamics are the physical stochastic process that takes place on the substrate of a fabricated chromophore network, a light source (e.g., QD-LED, laser) can excite the chromophore network to initialize the stochastic process and a

photon detector (e.g., SPAD) can detect the fluorescence photons from the chromophore network to read from the stochastic process. The exciton dynamics in the excited chromophore networks in the fabricated ensemble physically implement different realizations of the stochastic process in parallel, and the unique advantage of the readout method lies in its capability of measuring the stochastic behavior of the stochastic process by observing its individual realizations because each detected fluorescence photon is a random sample generated from a single chromophore network. While different ways of using this molecular photonic device exist with vast potential applications, this dissertation focuses on using it as a programmable entropy source to directly generate random samples from virtually arbitrary distributions, and illustrates its function in two applications: 1) fluorescent taggants and 2) stochastic computing.

1.3 Application I: fluorescent taggants

Fluorescent taggants have been widely used for labeling and identification applications, and they are usually made of organic dyes, quantum dots, metal complexes, etc [19-25]. Because fluorescent taggants with different compositions absorb or emit light in distinct wavelength regions, their spectral characteristics are often used as their optical signatures. While a few taggants can often be selected for reliable spectral discrimination, this approach suffers from poor scalability when target applications require a larger taggant library. The spectra of fluorescent materials are not easy to alter,

which greatly constrains the number of resolvable taggants that can be made of finite available fluorescent materials.

Within this context, we explore using the time-resolved fluorescence signal of a taggant as an alternative way to encode its optical signature. Based on the two observations that 1) a RET network is an entropy source that generates true random numbers from a phase-type distribution that is unique to the network and hence its signature and 2) RET networks can generate virtually arbitrary temporal distributions, we propose RET network based fluorescent taggants, which can potentially bring a significantly larger coding capacity and flexibility to taggant design.

On the detection side, time-resolved photon detection with a single pair of interrogation and detection wavelengths facilitates the detection of all taggants when the signatures are encoded in the time domain. Meanwhile, the process of taggant identification becomes an estimation problem where observed random numbers are used to estimate the unknown generating distribution, and statistical methods such as Maximum Likelihood Estimation (MLE) enable a robust and convenient taggant identification even under low light conditions and are able to resolve a mixture of taggants in multiplex detection. With these unique advantages, the fluorescent taggants with temporal signatures have great potential for both in situ and Lidar applications.

1.4 Application II: stochastic computing

Probabilistic algorithms and statistical methods are increasingly used in a wide variety of fields such as computer vision, robot/drone control, data mining, global health, computational biology, and economics. Based on generating samples from parameterized probability distributions, probabilistic algorithms are the only viable approach to the exact solution of many important classes of problems (e.g., high-dimensional inference, rare event simulation), and offer the potential to create generalized frameworks for broad applications.

Despite the theoretical advances in statistics and probabilistic machine learning, the fundamental mismatch persists between the deterministic hardware that traditional computers use and the stochastic nature of probabilistic algorithms. Modern computers largely take a deterministic approach to computation and are designed with deterministic algorithms and transistor functionality in mind; recent challenges in CMOS scaling reveal practical limits on performance.

Therefore, the challenge we propose is to develop new hardware that directly supports a wide variety of probabilistic algorithms [26]. Based on the observation that novel probabilistic functional units can be created using RET networks to approximate arbitrary probabilistic behavior and generate random samples from general distributions [18], this dissertation takes the first steps toward meeting this challenge by exploiting the physical properties of the molecular-scale photonic device.

We introduce the concept of a RET-based Sampling Unit (RSU), a hybrid CMOS/RET functional unit that generates samples from parameterized distributions. An RSU specializes the calculation of distribution parameters in CMOS and uses RET to generate samples from a parameterized distribution in only a few nanoseconds. In this work, we focus on accelerating the MCMC solver for MRF inference problems, and introduce RSU-G, a Gibbs sampling unit based on the RET-based exponential sampling units. Our specific RSU-G unit supports first-order MRFs with a smoothness-based prior, which includes many image processing applications (e.g., image segmentation, motion estimation, stereo vision).

The proposed molecular-scale optical Gibbs sampling unit (RSU-G) can be integrated into a processor / GPU as specialized functional units or organized as a discrete accelerator. Emulation-based evaluation of two computer vision applications for HD images reveal that an RSU augmented GPU provides speedups over a GPU of 3 and 16. Analytic evaluation shows a discrete accelerator that is limited by 336 GB/s DRAM produces speedups of 21 and 54 versus the GPU implementations.

The rest of the dissertation is organized as follows. Chapter 2 describes the exciton dynamics in a molecular-scale RET network, how they physically implements a CTMC, and the accurate fabrication via DNA self-assembly. Chapter 3 discusses using the molecular-scale CTMC as a programmable entropy source to directly generate random samples from virtually arbitrary distributions. Chapters 4 and 5 respectively

demonstrate the functions of RET networks as an entropy source in two applications: 1) fluorescent taggants and 2) stochastic computing, and Chapter 6 discusses other potential applications. Finally, Chapter 7 concludes the dissertation.

2. Resonance Energy Transfer Network Implemented Stochastic Process

The Poisson process and Exponential distribution naturally exist in nanoscale processes (e.g., chemical reactions, intermolecular energy transfer, luminescence, and electrostatics) and, more importantly, can be flexibly convolved and mixed to form different instances of continuous-time Markov chains, thus they provide unique opportunities for the molecular-scale physical implementation of stochastic processes.

Resonance Energy Transfer is a well-studied mechanism describing energy transfer between two chromophores, and commonly used to measure nanoscale distances and elucidate molecular structures and interactions in biology and chemistry. The RET transfer between two chromophores is exponentially distributed in the time domain, and therefore a molecular-scale RET network is a physical analog of a CTMC, where a direct mapping exists between the physical geometry of the chromophore network and the transition matrix of the CTMC. An ensemble of RET networks can be conveniently and economically fabricated in massive quantities (billions or more) using DNA self-assembly with sub-nanometer precision. Hence RET technology becomes a natural substrate for implementing and programming CTMCs.

The RET network implemented stochastic process occurs on the nanosecond timescale, which is fast enough for practical applications including those that emphasize performance such as stochastic computing demonstrated in Chapter 6. Meanwhile, the operation of the molecular photonic device is based on using a light source (e.g., QD-

LED) to excite the chromophore network to initialize the CTMC and a single photon detector (e.g., SPAD) to detect fluorescence photons from the chromophore network to sample from the CTMC. The exciton dynamics in the excited chromophore networks in the fabricated ensemble physically implement different realizations of the stochastic process in parallel, and the readout method is able to measure the stochastic behavior of the stochastic process by observing its individual realizations because each detected fluorescence photon is a random sample generated from a single chromophore network.

2.1 Resonance energy transfer network between chromophores

A RET network can be built by placing multiple chromophores in a physical geometry where each chromophore may interact with the others through resonance energy transfer (Figure 1). A chromophore is a molecule that can absorb photons at a specific wavelength and reemit at a longer wavelength via fluorescence. However, when two chromophores are placed a few nanometers apart and their emission and excitation spectra overlap, energy transfer can occur between the two chromophores through a process called Resonance Energy Transfer (RET). RET is an energy transfer mechanism between two chromophores where the donor chromophore, initially in its excited state, transfers its energy to the acceptor chromophore through non-radiative dipole-dipole coupling (Figure 2) [17]. The time to RET transfer, after the donor is excited, follows an exponential distribution between a chromophore pair. The fundamental parameters that

govern the transfer rates between the chromophores in a RET network are defined by the molecules we choose to create the network and the separation between them.

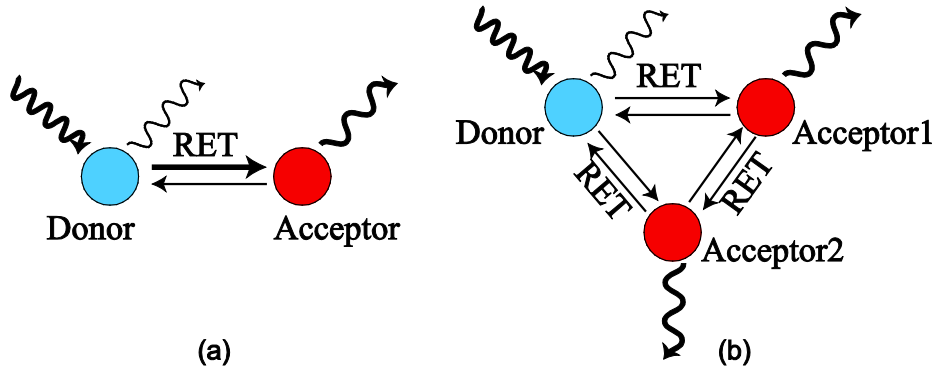

Figure 1: A resonance energy transfer network with (a) two chromophores and (b) three chromophores.

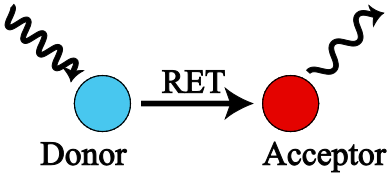

Figure 2: Resonance Energy Transfer.

The transfer rate of the RET process between a chromophore pair, first derived by Förster, is:

$$k_{RET} = \frac{3}{2}\frac{k^2}{\tau_D^0}\left(\frac{R_0}{r}\right)^6, \tag{1}$$

where τ_D^0 is the intrinsic fluorescence lifetime of the donor, k^2 is the mutual orientation of the chromophore pair (within the range between 0 and 4, for free-rotating chromophores 2/3 is used), r is the distance between the chromophore pair, and R_0 is the Förster radius for $k^2 = \frac{2}{3}$, i.e., the distance at which the transfer efficiency is 50%. The

Förster radius of a chromophore pair mainly depends on the properties of the two chromophores:

$$R_0 = 0.2108\left[\frac{2}{3}\Phi_D^0 n^{-4}\int_0^\infty I_D(\lambda)\varepsilon_A(\lambda)\lambda^4 d\lambda\right]^{1/6}, \quad (2)$$

where Φ_D^0 is the fluorescence quantum yield of the donor in the absence of transfer, n is the average refractive index of the medium within the wavelength range of significant spectral overlap, $I_D(\lambda)$ is the normalized fluorescence spectrum of the donor, $\varepsilon_A(\lambda)$ is the molar absorption coefficient of the acceptor and λ is wavelength.

The intrinsic fluorescence lifetime τ^0 of a chromophore is determined by the rates of all the intrinsic relaxation pathways including both radiative pathway (i.e., fluorescence) and nonradiative pathways. In the presence of RET to an acceptor chromophore, another relaxation pathway exists with the rate k_{RET} in Eq. (1). The exciton relaxation through each pathway is an exponentially distributed random variable in the time domain, and, as a result, the de-excitation of the chromophore is also exponentially distributed. Between a RET pair, the excited state lifetime of the donor chromophore is shown in Eq. (3), and the transfer efficiency is shown in Eq. (4).

$$\tau_D = \frac{1}{1/\tau_D^0 + k_{RET}} \quad (3)$$

$$TE = \frac{k_{RET}}{1/\tau_D^0 + k_{RET}} = \frac{1}{1+(r/R_0)^6} \quad (4)$$

2.2 Exciton dynamics of a RET network: continuous-time Markov chain

After a chromophore is excited in a RET network, the exciton dynamics in the RET network comprise of the sequence of RET transfers and the sojourn time of each RET transfer until it leaves the network through exciton relaxation, which are an absorbing Continuous Time Markov Chain [18]. CTMC is a continuous-time stochastic process with a finite or countable state space S, in which the time spent in each state is exponentially distributed. The Markov property of a CTMC means that the conditional probability distribution of future states of the process (conditional on both past and present states) depends only on the present state and not on the sequence of events that preceded it. A CTMC is defined by its discrete state space S, a transition matrix Q that indicates the transition rate between each pair of states, and an initial probability distribution $\pi(0)$.

An absorbing CTMC has at least one absorbing state $S_{Ai}\ (i=1,\ldots,n)$, which only has incoming transition rates, and the probability of the system having transitioned into an absorbing state approximates 1 as time increases to infinity: $\lim_{t\to\infty} Prob\left(X(t)\in\{S_{Ai}, i=1,\ldots,n\}\right)=1$. Additionally, the absorption probability of each absorbing state $S_{Ai}\ (i=1,\ldots,n)$, i.e., the probability of the system transitioning into each specific absorbing state, is $P_{Ai}=Prob\left(X(\infty)=S_{Ai}\right)$, which depends on the initial probability distribution $\pi(0)$ and the transition matrix Q [27].

An important and unique feature of the CTMC that captures the exciton dynamics of a RET network is the direct mapping between the CTMC and the RET network. The state space of the CTMC is composed of transient states and absorbing states. Each transient state S_T corresponds to a specific chromophore being excited, and the transition rate between a pair of transient states is the RET transfer rate between the corresponding chromophore pair. Each absorbing state S_A corresponds to the exciton leaving the RET network through a specific relaxation pathway, and the transition rate between a transient state and an absorbing state is the decay rate of the corresponding relaxation pathway of the corresponding chromophore. (Note: It is assumed that an ultrashort pulse laser only excites the donor chromophore and at most one chromophore remains excited between the pair at any time, which is feasible from our experience with wavelength division multiplexing and fabricated RET networks.)

Figure 3(a) shows the exciton dynamics between a chromophore pair and its corresponding CTMC model 6(b). In the CTMC, each chromophore has a transient state S_T to indicate whether it is in its excited state, and the exciton decay through each intrinsic relaxation pathway of each chromophore is represented as an absorbing state S_A, i.e., S_{A1}: 'Donor Fluoresces', S_{A2}: 'Acceptor Fluoresces', S_{A3}: 'Donor Nonradiative Decay', S_{A4}: 'Acceptor Nonradiative Decay'. (Although not shown in Figure 3(a), nonradiative decay exists as an exciton relaxation pathway and is included in Figure

3(b).) The initial state vector $\pi(0)$ and the transition matrix Q of the CTMC are shown in Eq. (5) and Eq. (6).

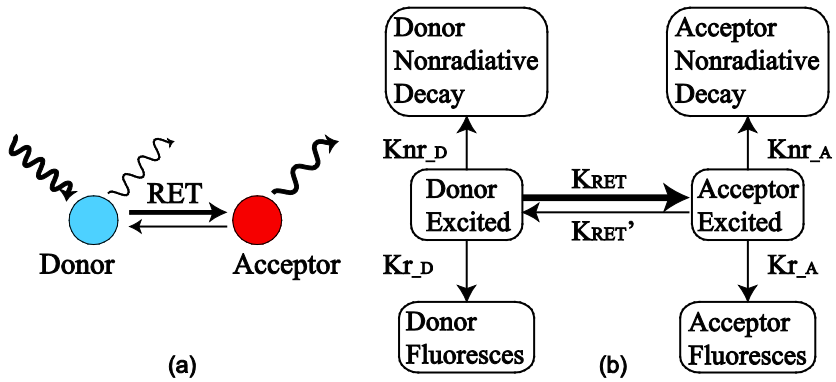

Figure 3: (a) The exciton dynamics between a RET pair and (b) its corresponding CTMC.

$$\pi(0) = [1,0,0,0,0,0] \tag{5}$$

$$Q = \begin{bmatrix} \begin{matrix} S_{TD} & S_{TA} \end{matrix} & \begin{matrix} S_{A1} & S_{A2} & S_{A3} & S_{A4} \end{matrix} \\ \begin{bmatrix} -K_{RET} - K_{r\_D} - K_{nr\_D} & K_{RET} \\ K_{RET'} & -K_{RET'} - K_{r\_A} - K_{nr\_A} \end{bmatrix} & \begin{bmatrix} K_{r\_D} & 0 & K_{nr\_D} & 0 \\ 0 & K_{r\_A} & 0 & K_{nr\_A} \end{bmatrix} \\ 0_{4X2} & 0_{4X4} \end{bmatrix} \tag{6}$$

For a larger RET network with more than two chromophores, its CTMC will simply enclose more states and the transition rates between them following the direct mapping (Figure 4). Therefore, a RET network is a direct and physical analog of a CTMC, which provides unique opportunities for physically implementing and programming molecular-scale CTMCs. The model specifications of a CTMC, such as state space and transition rates, can be explicitly implemented through the physical specifications of a RET network such as network size, RET transfer rates and exciton

decay rates. With the fabrication method based on DNA self-assembly, the physical specifications of a RET network can be precisely programmed by the physical geometry of a chromophore network; network size and exciton decay rates are directly controlled through the number and types of chosen chromophores, and RET transfer rates are controlled through the parameters in the Förster equation (Eq. (1)) such as distance. This direct and intuitive approach to physically implementing and programming CTMCs at the molecular scale is unprecedented.

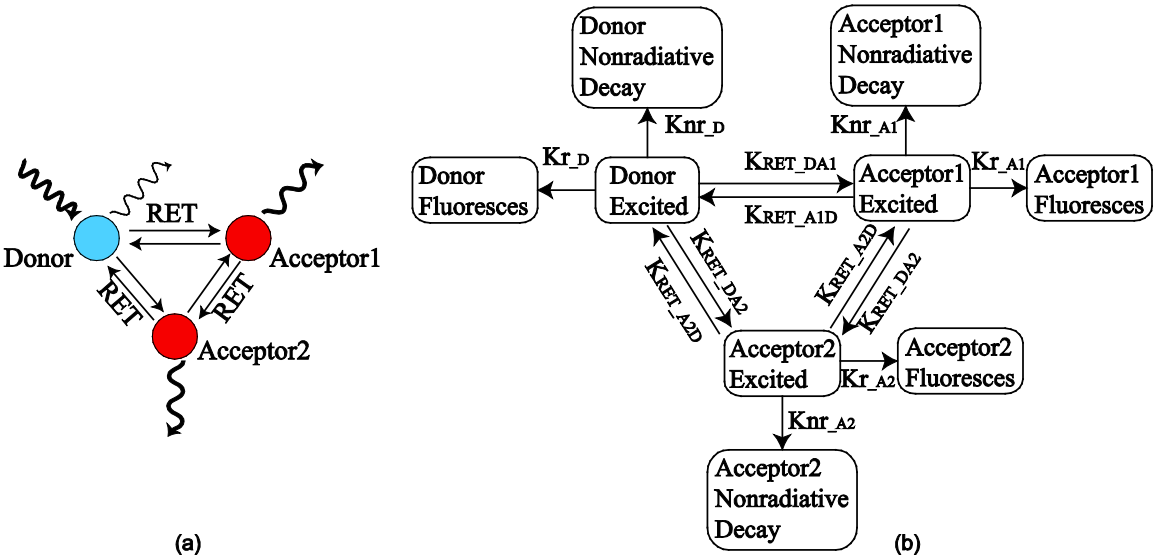

Figure 4: (a) The exciton dynamics in a 3-chromophore RET network and (b) its corresponding CTMC.

2.3 RET network fabrication via DNA self-assembly

Because of the inverse sixth-power distance dependence of RET, accurate fabrication of a chromophore network in terms of the distance between each chromophore pair becomes crucial for physically implementing specified exciton

dynamics and the CTMC. Extensively investigated as an economical approach to molecular-scale fabrication with subnanometer precision, DNA self-assembly is an ideal method for precisely fabricating chromophore networks.

Self-assembly is a process in which unorganized components form an organized system due to the local interactions among the pre-existing components without external forces. The local interactions responsible for self-assembly are weak forces, including π-π, Van der Waals and hydrogen bonds, which is in contrast to stronger forces in conventional chemical reactions: covalent, ionic and metallic bonds. Another important property of self-assembly is its thermodynamic stability. With weak local interactions as the driving forces in self-assembly, pre-existing components lead to a more organized system with a lower Gibbs free energy. Thus, locating the thermodynamic minimums in the energy landscape of configuration space is important for analyzing a self-assembly process. In addition, due to the weak nature of the interactions in self-assembly, minor changes of the environmental variables (e.g., temperature) can lead to significant deviation from the previous energy landscape, and therefore change the result self-assembled structure. This can be significantly useful in the sense that the control over external environment can be used to direct the self-assembly process.

Compared with other molecular self-assembly methods [28, 29], DNA self-assembly has unique advantages for fabricating RET networks such as the capability of making irregular and asymmetrical geometries with sub-nanometer precision, low cost

and high throughput. DNA self-assembly is an instance of molecular self-assembly where deoxyribonucleic acid (DNA) single strands are the pre-existing components for self-assembly. Single strands are mostly likely to bind to their complementary strands through hydrogen bonds following the rules of Watson-Crick base pairing (A with T and C with G). Based on this property, we can make specific sequence designs for different strands to let them go to specific positions when forming a desired structure, which minimizes the Gibbs free energy of the assembled structure. The current implementation of DNA self-assembly can be categorized into two types: folding-based self-assembly (i.e., DNA origami) [30] and hierarchical self-assembly [31]. Intensive research has been done in the field of synthesizing various planar and 3D DNA structures using DNA self-assembly for different applications, such as nanoelectronics, biomedicine, etc. The advantages of DNA self-assembly as a bottom-up fabrication approach are becoming increasingly appealing as traditional top-down fabrication methods are reaching their fundamental physics limits.

In this dissertation, chromophore networks are fabricated using hierarchical DNA self-assembly, where small building blocks are built first from the single strands and then assemble into a larger grid structure [31-34]. In this method, 9 single strands form a DNA cross-shaped motif with unique sticky ends extending out in 4 directions. 4 (or 16) such motifs with distinct sticky ends then assemble into a 2X2 (or 4X4) grid structure (Figure 5). The two assembly steps are completed during different annealing

processes which are detailed elsewhere [31-34]. Because of the hierarchical nature of the assembly process, it is possible to reuse single strands for different motifs, which reduces the fabrication cost and makes the assembly approach scalable. The 4-tile DNA grid is 40X40 nm^2 in size, and the 9 single strands of each tile contain 392 nucleobases in total. Meanwhile, single strands can be conjugated with chromophores at arbitrary bases before they are assembled into motifs. The chromophore–DNA conjugation can be achieved by using a primary amino modifier group on Thymidine to attach an NHS ester-modified dye molecule. By using these conjugated single strands in the self-assembly process (Figure 6), the DNA grids become fully addressable and programmable, and a chromophore network can be accurately patterned on the final grids (Figure 5).

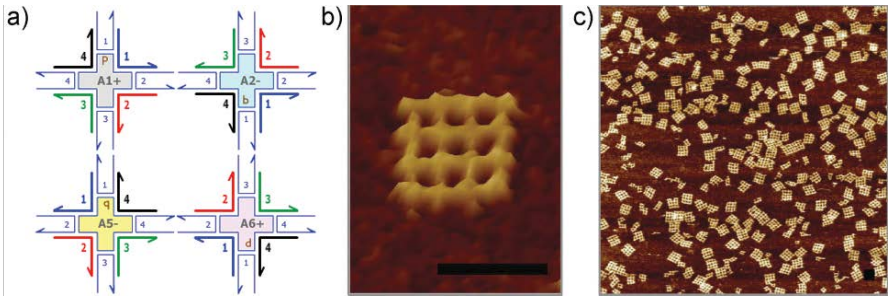

Figure 5: (a) Layout of a 2X2 DNA grid synthesized using hierarchical DNA self-assembly. (b)(c) AFM images of grid structures composed of 16 tiles. Scale bar: 60 nm [33].

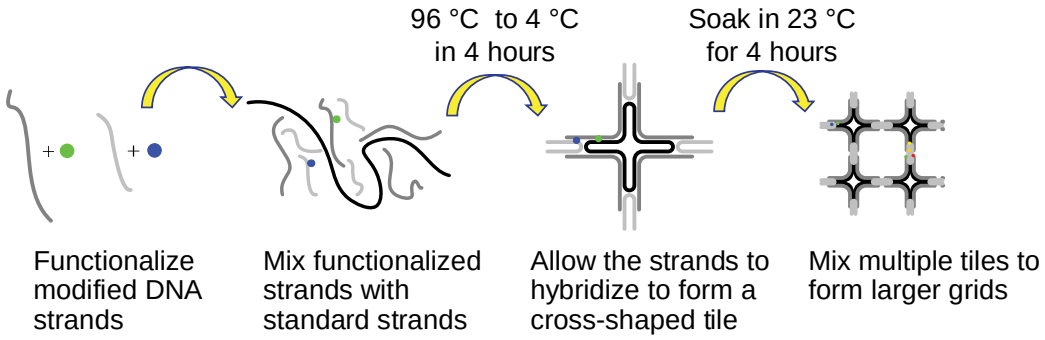

Figure 6: The fabrication process of a chromophore network based on hierarchical DNA self-assembly.

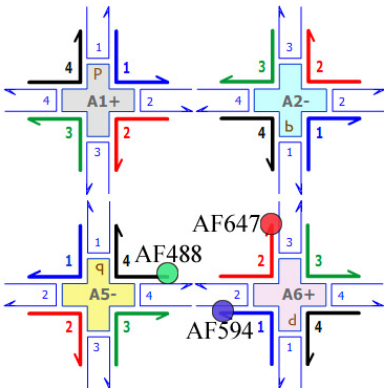

Figure 7: A chromophore network patterned on a 2X2 DNA grid.

Because chromophores can be conjugated with DNA strands at arbitrary nucleobases and each nucleobase is 0.33nm long, the fabrication of a chromophore network reaches subnanometer precision, which is sufficient given common Förster radii around 5~10nm. Meanwhile, this makes it feasible and convenient to fabricate irregular and asymmetrical network geometries which is difficult for other self-assembly methods. Other unique advantages of the fabrication method include massive parallelism and low cost. With commercially available chromophores and custom-designed DNA strands, the fabrication process can produce a manufacturing scale of

more than 10^{13} grids in 60μL within only a few hours, and the ensemble costs less than $1 USD. The product of chromophore–DNA conjugation can be analyzed and purified by using high-performance liquid chromatography (HPLC), and the yield of completely assembled grids and their structures can be characterized and observed by using an atomic force microscope (AFM).

It should be noted that organic dyes are not the only option for making RET networks, and alternative options include other commonly used fluorescent molecules such as fluorescent proteins and quantum dots (QDs). However, organic dyes are relatively more suitable for building RET networks for the following reasons.

First of all, RET networks with different structures and transfer rates are required to physically implement distinct stochastic processes, and organic dyes, among the common fluorescent molecules, are the most suitable and flexible choice for making RET pairs [35]. There is a large and diverse collection of commercial functionalized organic dyes, and their optical properties (e.g., absorption/emission spectra, extinction coefficient, quantum yield) and RET properties as donors and acceptors (i.e., Förster radius) are well characterized. These synthetic dyes are continually improved in terms of photostability, solubility and conjugation strategies. Fluorescent proteins are very similar to organic dyes in terms of their photophysics and optical properties and also used in RET applications, but they have larger sizes (~25kD) and lower photostability than organic dyes and their selection is much smaller [36].

Organic dyes have relatively narrower absorption bands, and it is feasible to selectively excite individual dyes to induce a sequence of RET transfers among multiple different dyes. Additionally, the large selection of commercially available organic dyes cover a wide spectral range from UV to NIR, and they often have relatively shorter Stokes shift. Thus RET pairs can be flexibly cascaded and organized to make different RET networks. In contrast, QDs have broad absorption bands and their absorption increases toward shorter wavelengths [35]. This property has limited QDs to the donor position in their few RET applications, which makes it difficult to only use QDs to create different RET networks. Meanwhile, the relatively larger size of QDs and their surface coating often make them less efficient in RET transfers compared with organic dyes. QDs are generally not recommended for RET applications.

More importantly, organic dyes typically exhibit mono-exponential decays, while QDs often exhibit multi-exponential decays [35]. As described in the previous section, the CTMC model of the exciton dynamics is based on the exponential distributions of RET transfers and fluorescence. Multi-exponential distributions would invalidate the direct mapping between a chromophore network and a CTMC and make it more challenging to physically implement the stochastic process.

2.4 Practical aspects of using the RET network implemented CTMC

The RET network implemented CTMC is compatible with practical applications of photonics and optoelectronics as a molecular photonic device in terms of input/output interface and speed.

Because exciton dynamics are the physical stochastic process that takes place on the substrate of a fabricated chromophore network, the input/output interface of the RET network should be based on a light source and a photon detector, and the specific implementation depends on whether the molecular photonic device is used as a discrete component or integrated in a photonic/optoelectonic circuit. A light source (e.g., QD-LED, laser) can excite the chromophore network to initialize the stochastic process and a photon detector (e.g., SPAD) can detect the fluorescence photons from the chromophore network to read from the stochastic process. Once the light source and the photon detector simultaneously turn on, the exciton dynamics in the excited chromophore networks in the fabricated ensemble physically implement different realizations of the stochastic process in parallel, and each detected fluorescence photon is a random sample generated from a single chromophore network and corresponds to a single realization. Therefore, this setup not only achieves massively parallel (10^{13} and more) simulation of a specified CTMC, but also conveniently measures the stochastic behavior of the molecular-scale stochastic process by observing its individual realizations.

Meanwhile, the exciton dynamics between chromophores are composed of RET transfers and fluorescence that occur on the nanosecond timescale, thus the physical stochastic process is fast enough for most applications including those that emphasize performance. The clock rate of modern CPUs is in GigaHertz (GHz), and the timescales of a CMOS transistor and a RET network are comparable.

While this molecular photonic device can potentially perform different functions and bring vast applications, this dissertation focuses on using it as a programmable entropy source to directly generate random samples from virtually arbitrary distributions, and illustrates its function in two applications: 1) fluorescent taggants and 2) stochastic computing.

2.5 DNA as a substrate for programming molecular interactions

So far, this chapter has described the molecular-scale implementation of CTMCs based on programming molecular interactions through the RET transfers between chromophores. DNA has been commonly used as a substrate for programming molecular interactions, and this section briefly discusses this approach and its theoretical and practical difficulties in terms of implementing stochastic processes.

Despite its intrinsic stochasticity, most applications of DNA-based chemical reactions abstract away the reaction kinetics of the process as deterministic behaviors with the law of mass action, and programming DNA strands to implement general stochastic processes is constrained by the chemical reaction network (CRN) description

of a reaction process. Meanwhile, the implementation aspects of this process include slow processing, thermal annealing, aqueous solvents and inconvenient readout, which are often incompatible with practical applications.

Deoxyribonucleic acid (DNA) provides a viable approach to programming molecular interactions and has been extensively studied for a variety of applications over the past two decades after Leonard Adleman first demonstrated its ability to solve combinatorial problems such as the seven-point Hamiltonian path problem [37]. DNA strands are composed of nucleotides which are covalently linked to form polynucleotide chains. Each nucleotide is composed of a nitrogen-containing nucleobase—either cytosine (C), guanine (G), adenine (A), or thymine (T)—as well as a sugar called deoxyribose and a phosphate group. The nucleobases on two DNA strands are bound to each other by hydrogen bonds to form base pairs following the rules of Watson-Crick base pairing (A with T and C with G), and stack upon one another, which leads to the double helical structure of DNA molecules. By designing the sequences of DNA strands to control their degree of complementarity, the hybridization reaction between the strands can be programmed.

However, programming DNA strands to control their reaction kinetics has implementation aspects that are often incompatible with practical applications. Besides the complex sequence design, the reaction process has unfavorable features such as slow processing, thermal annealing and aqueous solvents. A common DNA self-

assembly/computing process takes from minutes to hours or even days, and requires thermal annealing procedures which are unjustifiable beyond proof-of-concept purposes. Meanwhile, reading or sampling from the DNA based stochastic process is difficult and requires analyzing individual molecules or extracting molecular counts. To analyze the deterministic behaviors of a DNA-based chemical system, the readout methods rely on unwieldy laboratory equipment such as an AFM to examine the structure of assembled products or a fluorometer to measure reactant concentrations, and require highly specialized expertise and operate on a similar time scale to the slow reaction process.

While the reaction between DNA strands is a stochastic process at the molecular scale, most applications abstract away their reaction kinetics as deterministic behaviors with the law of mass action and forgo the inherent stochasticity. The applications of this field are mainly inspired by self-assembly and chemical reaction network (CRN) theory. As an approach to bottom-up nanofabrication or performing logic functions, arbitrary 2D or 3D nanostructures can be accurately self-assembled by designing the sequences of interacting DNA strands [38-41]. Meanwhile, CRN has a vast literature of theory and becomes a natural and effective programming language for complex network behaviors in DNA strand displacement-based computing, and specified CRNs can be compiled into DNA sequences to physically implement deterministic functions such as Boolean logic gates and oscillators [42-45]. Although DNA strand displacement in the low copy

number regime has been theoretically studied as a stochastic CRN (SCRN) to demonstrate its computational capabilities by storing information in molecular counts [46, 47], the physical implementation would be challenging and sampling from the stochastic process would require extracting the molecular counts of reactants and become almost infeasible. More importantly, the stochasticity in the proposed SCRN is deemed as the source of error to be diminished rather than an entropy source to be utilized.

Further, programming DNA strands to physically implement general stochastic processes is constrained by the CRN description of a reaction process. While the current applications that program the stochastic reaction between DNA strands at the molecular scale rely on the deterministic abstraction of a CRN at the macroscopic scale, this approach would fail in applications that require or benefit from the implementation of general stochastic processes. The stochastic process of a DNA based SCRN is the set of counting processes described by its Chemical Master Equations (CMEs), and becomes a CTMC where each state comprises of the molecular counts of each species and each transition corresponds to the occurrence of a reaction [10]. The complex coupled reactions involving large molecular counts in the SCRN translate to the numerous states and their complex transitions in the CTMC; such a CTMC easily becomes infeasible to numerically specify and requires simulation for modeling and analysis, and mapping it to another specified CTMC by programming the strands would be implausible.

Although DNA-based single-molecular chemical reactions could in principle implement a CTMC by mapping its states to molecular structures under the assumption that certain reacting species are sufficiently abundant in the solution, this approach would be difficult to realize and use because it relies on analyzing individual molecules.

3. A Programmable Entropy Source For Generating True Random Numbers

As discussed in Chapter 2, the exciton dynamics in a RET network of chromophores can physically implement programmable molecular-scale CTMCs, and they are compatible with practical applications of photonics and optoelectronics as a molecular photonic device. While vast applications can be potentially enabled by using the molecular photonic device to perform different functions, this dissertation focuses on using this device as an entropy source to generate true random numbers from general distributions and its applications.

Random numbers are critical in a wide range of fields such as probabilistic algorithms, cryptography, and numerical simulations. Dependent on the way they are generated, random numbers fall in two categories: 1) pseudorandom numbers and 2) true random numbers [48]. Pseudorandom numbers are generated by a pseudorandom number generator (PRNG), which is a deterministic algorithm that produces a sequence of periodic numbers whose properties approximate those of random numbers. Therefore, they are deterministic number sequences with strong long-range correlations and can be fully determined when the initial state, or seed, of the generating algorithm is known. In contrast, true random numbers are generated by a true random number generator (TRNG), which operates by measuring a physical process that behaves in a fundamentally nondeterministic way. The physical process is the entropy source that provides the randomness in a TRNG, and common physical RNGs can be divided into

four categories according to their entropy sources: noise based RNGs, free running oscillator (FRO) RNGs, chaos RNGs and quantum RNGs.

The quality of randomness is crucial for applications such as cryptography and stochastic simulations and calculations, and the absence of true randomness can cause a security breach or an erroneous result. Although frequently used in practice, PRNGs are deterministic and unacceptable for such applications. While there exist a variety of implementations of physical RNGs using different physical processes, an important aspect of these RNGs is often neglected, which is the provability of randomness. Among the physical processes used in existing physical RNGs, only those based on quantum systems can be information theoretically proven to be truly random. However, FRO and noise RNGs are the most widely used especially in the semiconductor industry due to their easier implementation.

Meanwhile, existing RNGs only produce uniform random numbers in the form of binary bits by using the nondeterministic behavior of a physical process to generate two equally likely outcomes. In applications that need random numbers with more general distributions, techniques such as inverse transform are required to map the uniform random numbers to a target distribution. While exponentially distributed random events exist in certain quantum RNGs [49], it remains difficult to find a physical process that can be precisely controlled to directly generate random numbers from general distributions.

Within this context, the exciton dynamics in a RET network of chromophores is explored as an entropy source that utilizes molecular quantum electrodynamics to produce randomness and can be precisely programmed to generate random events with general distributions. The random events occur in the form of fluorescence photons in the time domain. To produce random numbers from the RET based entropy source, a single photon detector (e.g., SPAD) and a Time-Correlated Single Photon Counting (TCSPC) module can be used to measure the detection times of fluorescence photons. To our knowledge, this is the first entropy source that can be programmed and used to directly generate true random numbers from virtually arbitrary distributions.

3.1 RET network as a source of true randomness

A RET network is a molecular quantum system composed of multiple chromophores, and the exciton dynamics in such a network can generate true randomness from molecular quantum electrodynamics, based on which the probabilistic behavior of the network can be precisely controlled and characterized from first principles. The RET transfer in this dissertation is between chromophores placed a few nanometers apart, in which region the Förster theory applies and the transfer rate has the inverse sixth-power dependence on the inter-chromophore distance. Förster Resonance Energy Transfer (FRET) is a short-range RET mechanism and also a radiationless mechanism which does not involve photon emission and absorption. In contrast, radiative RET is a long-range RET mechanism and involves photons being

emitted and absorbed between a chromophore pair, and its transfer rate has an inverse second power dependence on the inter-chromophore distance. These radiationless and radiative RET mechanisms are the short- and long-range limits of a unified RET theory that originates from molecular quantum electrodynamics [50, 51]. These RET systems are based on weakly coupled dipole-dipole interactions, and their transfer rates can be derived by using the Fermi golden rule as a correlation function in the interaction. Therefore, the exciton dynamics in a RET network exploits quantum optical processes to generate true randomness, and the stochastic process of the exciton dynamics and the generated probability distribution can be precisely derived and controlled.

3.2 Time-resolved fluorescence of a RET network: phase-type distribution

The time-resolved fluorescence of a RET network has a phase-type distribution that is defined by the absorbing CTMC of the exciton dynamics in the network. Because the time to RET transfer, after the donor is excited, follows an exponential distribution between each chromophore pair and the sequence of RET transfers between an exciton entering and leaving (i.e., decaying) a chromophore network is a random process, the time to exciton decay follows a phase-type distribution [18]. The RET transfer between a chromophore pair with a transfer rate of k_{RET} physically implements a phase transition with a transition rate of $\lambda = k_{RET}$ in the phase-type distribution, and the geometry of the chromophore network controls how these phases are convolved and mixed to form the phase-type distribution. Specifically, the time to exciton decay in a RET network is the

time to absorption in the absorbing CTMC of its exciton dynamics, and the CTMC specifies the distribution of this phase-type random variable through its transition matrix Q which contains the RET transfer rate between each chromophore pair and the decay rate of each relaxation pathway.

Consider the RET pair in Figure 3 for simplicity. An Alexa Fluor 488 dye and an Alexa Fluor 594 dye are chosen as the donor and acceptor respectively, which are placed 10nm apart. The state probabilities of the four absorbing states in the CTMC, $\pi_{SA}(t)=\left[\pi_{SA1}(t),\pi_{SA2}(t),\pi_{SA3}(t),\pi_{SA4}(t)\right]$ (S_{A1}: 'Donor Fluoresces', S_{A2}: 'Acceptor Fluoresces', S_{A3}: 'Donor Nonradiative Decay', S_{A4}: 'Acceptor Nonradiative Decay'), are shown in Figure 8(a) and their sum monotonically approximates 1 as the input exciton is increasingly likely to have decayed as time passes. Specifically, the blue and red solid curves in Figure 8(a) correspond to the fluorescence of the two chromophores. By taking the derivative of these two curves and normalizing each, the conditional probability density function (PDF) of the time to fluorescence (TTF) from each chromophore, $f_T(t \mid X(\infty)=S_{A1})$ and $f_T(t \mid X(\infty)=S_{A2})$, can be derived (see Figure 8(b)), which are phase-type distributions. This method can be generalized to larger RET networks with more chromophores.

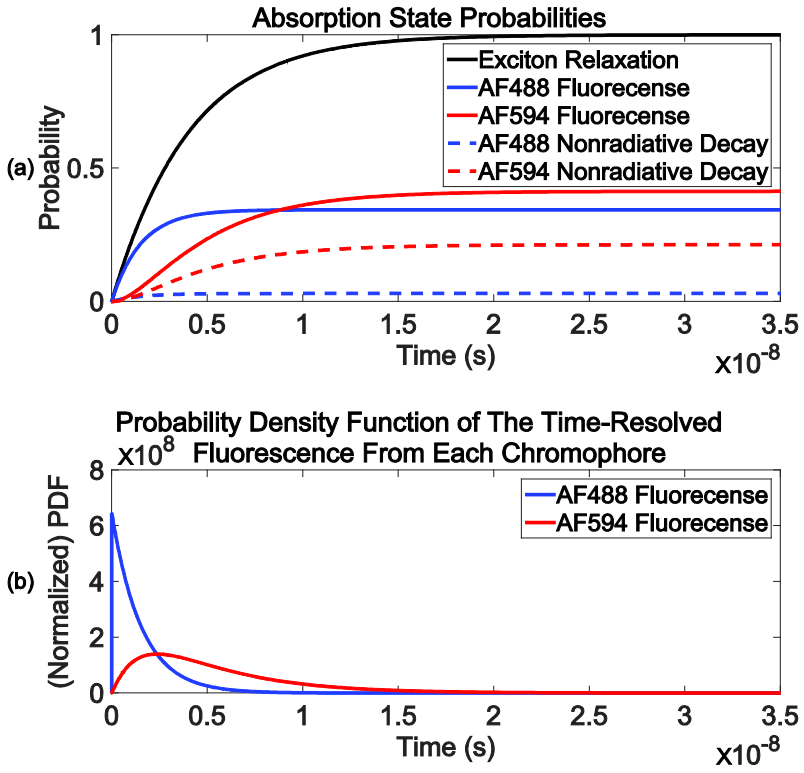

Figure 8: (a) The state probabilities of the four absorbing states in the CTMC in Figure 6(b) and their sum and (b) the conditional PDF of the time-resolved fluorescence from each chromophore.

3.3 Generating true random numbers from general distributions

Because the fluorescence photons emitted from a RET network follow a phase-type distribution that is configured by its CTMC and phase-type distributions can approximate any positive-valued distribution to arbitrary precision, it becomes theoretically feasible to use RET networks as a programmable entropy source to generate true random numbers from general distributions. The physical geometry of a chromophore network is designed so that the phase-type distribution of its time-resolved fluorescence approximates the target distribution. To generate random

numbers from a RET network, a light source (e.g., QD-LED, laser) excites the source chromophore in the network and a single photon detector (e.g., SPAD) and a high-resolution photon timing module (e.g., TCSPC) respectively detects and times the fluorescence photons from the designated emitter chromophore in the network.

3.3.1 Theoretical RET network: a universal entropy source

A RET network is a programmable entropy source that is theoretically capable of generating random events in the form of fluorescence photons from general distributions. Based on the direct mapping between a RET network and the phase-type distribution configured by its absorbing CTMC, a RET network can be designed in terms of network size and rate constants to generate any phase-type distribution. Phase-type distributions and CTMCs are often used to approximate general distributions to analyze non-Markovian systems because they have a vast literature of theory and their Markovian properties offer easier and even analytically tractable solutions. In theory, phase-type distributions form a dense set in the field of all positive-valued distributions and any positive-valued discrete or continuous distribution can be approximated with a phase-type distribution to arbitrary precision [52, 53]. The methods of approximating a general distribution with a phase-type distribution have been well investigated, and the two common approaches are based on (1) Moment-matching [54] and (2) Minimization of a difference (e.g., Kullback-Leibler Divergence) [55]. Therefore, a RET network can be

specified in terms of network size and rate constants to implement a phase-type distribution that asymptotically generates random events from a general distribution.

The design of the RET network should establish the direct mapping between itself and the absorbing CTMC that configures the phase-type distribution. The time to fluorescence (TTF), i.e., the time between an exciton enters the RET network from its source chromophore and leaves from its emitter chromophore(s) through fluorescence, is the time to absorption in the CTMC, which should have the phase-type distribution. The source chromophore should correspond to the initial state of the phase-type distribution, and the fluorescence of the emitter chromophore(s) should correspond to the absorbing state(s) of the phase-type distribution. The size (i.e., number of chromophores) of the RET network should equal the number of transient states in the absorbing CTMC, and the RET transfer rate between each chromophore pair should equal (or scale with) the transition rate between the two corresponding transient states. The radiative decay rate of the emitter chromophore(s) should equal (or scale with) the transition rate between its corresponding transient state and absorbing state for fluorescence, and the decay rates of all the other exciton relaxation pathways in the RET network should be significantly slower compared with the aforementioned transition rates so that their effect on the phase-type distribution becomes negligible.

3.3.2 Practical implementation of a RET network based TRNG

To generate random numbers from a RET network, a light source (e.g., QD-LED, laser) is needed to send a periodic train of ultrashort pulses to excite the source chromophore in the RET network, and a single photon detector (e.g., SPAD) detects the fluorescence photons emitted from the emitter chromophore of the RET network and a high-resolution single photon timing module such as Time-Correlated Single Photon Counting (TCSPC) measures the detection times of these photons relative to the excitation pulses with a timing resolution as high as several picoseconds (Figure 9). The photon detection times are the random numbers that are generated from the phase-type distribution encoded in the RET network.

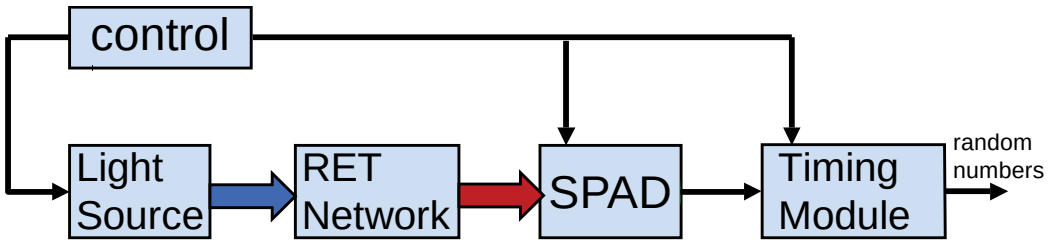

Figure 9: The diagram of a RET network based TRNG.

With DNA self-assembly, many copies of a RET network are fabricated and exist as an ensemble. When measuring fluorescence photons from the ensemble, their detection times follow the intrinsic phase-type distribution of the RET network if the system operates in a low-light condition for TCSPC to correctly function [56]. It needs to be ensured that the fluorescence intensity received by the SPAD is so low that the probability of detecting a photon after each excitation pulse is considerably lower than 1.

When the RET network ensemble is periodically excited by its light source, the TCSPC system registers a single photon in some excitation cycles and fails to register any photon in almost all other cycles, and the probability of detecting more than one photon in a cycle is extremely rare. Under this condition, the probability of detecting a photon at a certain time in a cycle is proportional to the time-resolved fluorescence intensity, and the histogram of the detection times relative to their excitation pulses approximates the phase-type distribution. In practice, the probability of detecting a photon per cycle (i.e., count rate) is controlled to be between 1~5% to maintain the single photon statistics. Otherwise, a high fluorescence intensity would cause the 'pile-up' effect and distort the distribution of detection times due to the 'dead' time of the SPAD. After detecting a photon, a SPAD enters a state where it cannot detect other photons, and the state typically lasts a few nanoseconds, i.e., its dead time. When the fluorescence intensity is high and multiple photon detections are expected in an excitation cycle, only the first photon will be detected and the rest will be dropped due to the SPAD's dead time.

The specific implementation of the RET network based TRNG varies between applications. When RET networks are used as fluorescent taggants (Chapter 4), a macroscale light source (e.g., laser) and detection components, including a SPAD and a TCSPC, can interrogate and identify a RET network from a distance (meters or even kilometers). When RET networks are used to generate random numbers for probabilistic computing (Chapter 5), nanoscale light sources (e.g., QD-LEDs), a SPAD and a RET

network are integrated as a nanoscale optoelectronic device that is used as a functional unit for sampling.

While theoretical RET networks can be designed to generate random numbers from virtually arbitrary distributions, the practical limits of chromophore networks may constrain the RET networks and distributions that can be fabricated, and it can be difficult to find a chromophore network to physically implement a theoretical RET network when the size and complexity of the network increases. In practice, the maximum network size is limited by the number of chromophores that can be conjugated on a DNA grid. While the 4-tile DNA grid used for fabricating chromophore networks in Chapter 2 has 392 nucleobases in total, the virus M13mp18 chosen as the scaffold for DNA origami contains 7249 nucleobases [30] and represents the largest scale of current DNA self-assembly. Because inter-chromophore distances are used to control RET transfer rates, only a small fraction of nucleobases are often available for dye conjugation, and it can be difficult to locate these nucleobases for a complex RET network. While the RET transfer rate can be individually specified for each RET pair in theory, this is usually difficult to achieve for a chromophore network in practice. Because chromophores are placed on a 2D DNA grid, moving the position of a chromophore is likely to change its RET transfer rates with multiple surrounding chromophores. Meanwhile, although the transitions between transient states in a CTMC are allowed to form loops with high probabilities, the probability of successive RET

transfers forming a loop is minimal due to the energy loss (i.e., Stokes shift) at each chromophore. In addition to network geometry, the finite set of commercially available organic dyes also constrains the RET networks and distributions that can be actually fabricated. Despite the large selection of commercially available organic dyes that cover a wide spectral range from UV to NIR and have diverse RET properties as donors and acceptors, choosing the chromophore types to constitute a RET network may still be challenging in practice. While it is relatively easy to find two chromophores to implement a single transfer rate, it is much more difficult to assign multiple (>2) chromophores so that their structure simultaneously satisfies the specified transfer rate between each chromophore pair, and the challenge is convolved with the physical geometry of the chromophore network. In addition, the back transfer rate between a chromophore pair is often nonzero and sometime non-negligible due to the small overlap between the acceptor's emission spectrum and the donor's absorption spectrum. While the dominant RET transfer rate between the pair can be controlled by parameters such as their distance, the back transfer rate between the pair is also affected.

3.3.2.1 Using chromophores wires to generate acyclic phase-type distributions

Although the question if a chromophore network can be designed and fabricated using organic dyes to physically generate an arbitrary phase-type distribution is an NP problem and does not have an easy and general solution, the types of phase-type distributions commonly used to approximate general continuous distributions have

simpler structures and are suitable for a chromophore network to generate. To approximate a general continuous distribution with a phase-type distribution is essentially an optimization problem with an infinite search space. In practice, this problem is often made manageable by limiting the search in a class of phase-type distributions with a simpler fixed structure [54, 55, 57-61]. They are usually acyclic phase-type (APH) distributions such as Erlang-Exponential or hyper-Erlang distributions with a fixed number of phases and branches, and the rates in the distribution are optimized to minimize the divergence between the APH distribution and the general distribution. Because of its acyclic structure, it becomes more feasible and convenient to design a chromophore network to generate the phase-type distribution. The RET transfer between a chromophore pair should physically implement a phase transition in the phase-type distribution, and the chromophore pairs should be cascaded in a wire geometry to minimize the transfer rates between non-adjacent chromophores. Unless otherwise required, chromophore pairs should be chosen so that their back transfer rates are negligible, and the intrinsic decay rates of chromophores should be significantly slower than the RET transfer rates in the chromophore wire. Therefore, the APH distributions that are commonly used to approximate continuous distributions are feasible and convenient for chromophore wires to generate.

However, the performance of these phase-type approximation methods varies between distributions in practice. While the approximation usually performs well for many regular continuous distributions such as Gaussian and Weibull with only a few phases, it may require an excess number (hundreds or thousands) of phases for 'bad-shaped' distributions such as density functions with discontinuities (e.g., uniform distribution) and complex distributions from observed data. Meanwhile, discrete distributions are not directly addressed in these phase-type approximation methods. It easily becomes impractical to use a single chromophore network to generate these distributions and different strategies should be considered for them.

3.3.2.2 Delayed CTMC (d-CTMC) and interval phase-type (IPH) approximation

For distributions with discrete components such as fixed delays and irregular distributions with complex shapes, conventional phase-type approximation usually requires an extreme amount of states and produces unsatisfactory result. Alternative phase-type approximation methods have been investigated, among which an approach called interval phase-type (IPH) approximation has unique advantages by introducing discrete-time events into the approximation and is suitable for RET networks to implement in practice [62].

IPH approximation uses piecewise phase-type distributions to approximate a given distribution, where the support of the given distribution is divided into multiple intervals and each phase-type distribution begins after a fixed time delay and

approximates the given distribution in a single interval. This method can manage distributions with a bounded support such as a delayed distribution (lower-bounded) or a uniform distribution (upper-bounded) and complex shaped density functions. For these distributions, IPH approximation only requires a moderate number of states and achieves much better result than conventional PH approximation.

Meanwhile, a TRNG based on this approach can be physically implemented by modifying the scheme of generating random numbers from RET networks. The implementation is directly based on the way piecewise phase-type distributions are delayed and composed to approximate a given distribution. Multiple chromophore networks are fabricated, and each one has an associated time delay and generates a phase-type distribution for an interval in the IPH approximation for the specified distribution. To generate random numbers from the specified distribution, the chromophore networks are excited sequentially in order of their corresponding intervals. When a chromophore network is excited, it is to be observed whether a detected fluorescence photon falls in its corresponding time interval. If it falls in its interval, a random number is generated which is the sum of the photon detection time within this interval and the delay associated with this interval. If the detected fluorescence photon is beyond its interval, a random number is not generated and the system proceeds to the chromophore network corresponding to the next time interval until a random number is eventually generated.

3.3.2.3 Discrete sampler based on competing exponential random variables

In addition to continuous distributions, generating random numbers (or samples) from discrete distributions are often necessary and can be conveniently implemented by RET networks based on the property of competing exponential random variables.

As mentioned earlier, the time to fluorescence (TTF) of a RET network is a phase-type distribution and the SPAD can be used to sample from the phase-type distribution. The exponential distribution is simply a one-stage phase-type distribution, thus it can be conveniently extracted from a RET network. Consider a single RET pair excited by one QD-LED at time $t = 0$. When the back transfer from the acceptor to the donor is negligible, the TTF of the donor is exponentially distributed with a decay rate λ_D (Eq. (7b)). When the SPAD is turned on to detect photons beginning at $t = 0$, the time to the first photon detection has an exponential distribution with a decay rate λ:

$$\begin{aligned} \lambda &= N * P_e P_{DF} P_d * \lambda_D \\ &= N * P_e P_d * k_{d_r}; \end{aligned} \tag{7a}$$

$$\begin{aligned} \lambda_D &= \lambda_{d0} + k_{RET} \\ &= k_{d_r} + k_{d_{nr}} + k_{RET}; \end{aligned} \tag{7b}$$

$$P_{DF} = \frac{k_{d_r}}{k_{d_r} + k_{d_{nr}} + k_{RET}}, \tag{7c}$$

where λ_D is the decay rate of the excited state of the donor chromophore in the presence of RET to the acceptor, which is the sum of its intrinsic excited state decay rate λ_{d0} and the RET transfer rate k_{RET}. P_{DF} is the probability that the donor fluoresces after excited,

which is the ratio between its radiative decay rate k_{d_r} and $\lambda_D = k_{d_r} + k_{d_{nr}} + k_{RET}$. P_e is the probability of the donor being excited by the QD-LED, and P_d is the photon detection efficiency of the SPAD. Many copies of the RET network exist in the fabricated ensemble, and N is the number of RET networks in the ensemble that can be excited. Although λ is independent of the RET transfer rate, the value of λ can still be engineered by changing the concentration of RET networks (N), the emission intensity of the QD-LED (P_e) and even the donor chromophore (k_{d_r}). In addition, as will be explained shortly, a set of such exponential distributions can be used to implement a specified discrete distribution, and only the relative ratio between their decay rates matter rather than their exact values.

It is noteworthy that the exponential distribution of the time to the first photon detection (Eq. (7)) relies on a higher fluorescence intensity received by the SPAD in contrast to the low fluorescence intensity TCSPC requires to correctly sample from the intrinsic phase-type distribution of the time to fluorescence of a RET network. To use the first photon detection to create the exponential random variable with a decay rate λ (Eq. (7)) different from the decay rate λ_D of the donor chromophore, the fluorescence intensity needs to be so high that at least one photon detection is expected in each excitation cycle and the SPAD only detects the first photon to generate samples. While the 'pile-up' effect is to be avoided for sampling from the intrinsic phase-type distribution of a RET network, this effect is utilized here to create an exponential

distribution that can be programmed through parameters such as the intensity of the light source and the concentration of the RET network.

From the above, an exponential sampler can be implemented using a RET network along with its light source and SPAD to generate random samples from an exponential distribution with a specified decay rate. Multiple such exponential samplers can be composed to create a discrete sampler that generate samples from discrete distributions based on the property of competing exponential random variables. Given M exponential random variables $X_i\left(i=1,\ldots,M\right)$ with decay rates $\lambda_i\left(i=1,\ldots,M\right)$, the probability the i th exponential random variable is the minimum among all the M random variables is [63]:

$$P\left(X_i=\min\left(X_1,\ldots,X_M\right)\right)=\frac{\lambda_i}{\sum_{j=1}^{M}\lambda_j}. \tag{8}$$

Consider a discrete random variable X with M possible outcomes $\left\{0,1,\ldots,M-1\right\}$ with the probabilities $\left\{P\left(X=0\right)=p_0,P\left(X=1\right)=p_1,\ldots,P\left(X=M-1\right)=p_{M-1}\right\}$ $\left(p_0+p_1+\ldots+p_{M-1}=1\right)$. The discrete random variable X can be implemented using M RET-based exponential random variables $X_0,X_1,...,X_{M-1}$ that correspond to the M outcomes $0,1,\ldots,M-1$. The decay rates of the M exponential random variables $X_0,X_1,...,X_{M-1}$ are such that their relative ratio equals the relative ratio between $p_0,p_1,...,p_{M-1}$, i.e., $\lambda_0:\lambda_1:...:\lambda_{M-1}=p_0:p_1:...:p_{M-1}$. Samples of the M -outcome discrete random variable X can be generated as follows. At time $t=0$, the QD-LEDs in

the M RET-based exponential samplers simultaneously send a delta pulse to excite their RET networks and all the SPADs are turned on simultaneously. The outcome corresponding to the RET network whose SPAD detects the first photon before all the other exponential samplers is considered a sample of X. The input to the discrete sampler is the signal to turn on the QD-LEDs and start the SPADS, while the output is the sample of X.

Therefore, RET networks are an entropy source that can be programmed to directly generate true random numbers from virtually arbitrary continuous and discrete distributions, and TRNGs based on this molecular-scale entropy source have different implementations. While RET networks can potentially benefit wide applications that rely on random numbers to perform various functions, we focus on two applications: 1) fluorescent taggants and 2) stochastic computing, which respectively leverage the two unique aspects of RET networks as an entropy source: 1) flexible programmability of probability distributions during fabrication and 2) efficient generation of random numbers through single photon detection.

4. Application in Fluorescent Taggants

Fluorescent taggants are widely used in labeling and identification applications, and they are often made of organic dyes, quantum dots and metal complexes [19-25]. Because different fluorescent materials absorb or emit light in distinct wavelength regions, their spectral characteristics are commonly used as their optical signatures for taggant identification and discrimination. However, it is difficult to use this approach to create a large taggant library because the spectra of fluorescent materials are difficult to control and the finite available fluorescent materials constrain the resolvable taggants that can be made.

Within this context, we explore using the time-resolved fluorescence of a taggant as an alternative way to encode its optical signature [64]. While time-resolved fluorescence has been considered for labeling applications in previous studies, it is so far limited to the intrinsic exponential decays of fluorescence, and hence constrained by the resolvable lifetimes of available chromophores [65-67]. In this thesis, RET networks are proposed to make temporally coded fluorescent taggants and bring a significantly larger coding capacity and flexibility to taggant design. As discussed in Chapter 3, a RET network is an entropy source that generates random events from a temporal phase-type distribution that is unique to the network and hence its signature. Because RET networks can generate virtually arbitrary time-resolved fluorescence signals, the coding capacity of the fluorescent taggants is considerably larger than their spectrally coded

counterpart. Meanwhile, this approach is not constrained by resolvable chromophores because the physical geometry of chromophore networks is leveraged to program their temporal signatures.

On the detection side, time-resolved photon detection with a single pair of interrogation and detection wavelengths facilitates the detection of all taggants when the signatures are encoded in the time domain. Meanwhile, the process of taggant identification becomes an estimation problem where observed random numbers are used to estimate the unknown generating distribution. The detected photons follow a temporal multinomial distribution in TCSPC, and statistical methods such as Maximum Likelihood Estimation (MLE) enable a robust and convenient taggant identification even under low light conditions. Further, a mixture of taggants, in multiplex detection, can also be formulated in MLE and resolved by the Expectation Maximization (EM) algorithm. With these unique advantages, the RET network based fluorescent taggants have great potential for both in situ and Lidar applications.

4.1 Fluorescent taggants with temporally coded signatures

Based on the two observations that 1) a RET network is an entropy source that generates random events in the form of fluorescence photons in the time domain from its intrinsic phase-type distribution and 2) RET networks can be programmed to create virtually arbitrary temporal phase-type distributions, RET network based fluorescent taggants are proposed and their optical signatures are encoded in their temporal phase-

type distributions. These temporally coded fluorescent taggants have a significantly larger coding capacity and flexibility than their spectrally coded counterpart. The methods of approximating a general distribution with a phase-type distribution have been discussed in Chapter 3, which provide guidance on designing a RET network given a target signature. Further, because network geometry is leveraged to create signatures in this approach, it is not constrained by resolvable chromophores unlike spectrally or lifetime coded fluorescent taggants. To illustrate this point, six different RET networks were designed using the same set of three dye molecules, i.e., Alexa Fluor 430, Alexa Fluor 594 and Alexa Fluor 750, and their temporal signatures were simulated and compared. The excitation and emission spectra of the three dyes are plotted in Figure 9, which shows that AF430 can be excited at a wavelength about 450nm with negligible interference with the other dyes and the fluorescence of AF750 can be measured at a wavelength about 780nm with negligible crosstalk from the other dyes. Each of the six RET networks adopts a wire geometry that contains one AF430 chromophore, one AF750 chromophore and a certain number ($N_i = i, i = 1, \ldots, 6$) of AF594 chromophores in between (see Figure 10). The distance between two adjacent chromophores equals their Förster radius R_0. The AF430 chromophore functions as an optical antenna and can be excited by a pulsed laser source, and the AF594 chromophores function as mediators that propagate excitons downwards, and the AF750 chromophore functions as an emitter and its time-resolved fluorescence is used as the temporal signature.

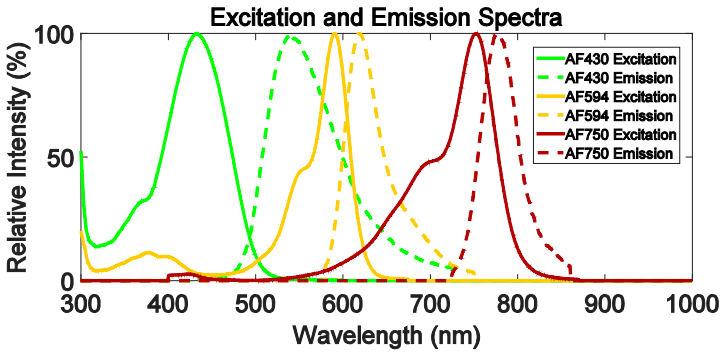

Figure 10: The excitation and emission spectra of AF430, AF594 and AF750.

Figure 11: The wire geometry of the six RET networks. The i'th RET network has $N_i = i$ AF594 chromophores between its AF430 and AF750 chromophores, and the distance between each adjacent chromophore pair equals their Förster radius.

In each RET network, RET transfer can occur between both adjacent and non-adjacent chromophores. Because an exciton may decay from any chromophore in a RET network, the end-to-end transfer efficiency from AF430 to AF750 varies as a function of the length of the chromophore wire and its pairwise transfer efficiencies. By building the CTMC model for each RET network, its temporal signature can be simulated by deriving the conditional PDF of the TTF from its AF750 chromophore (see Figure 12).

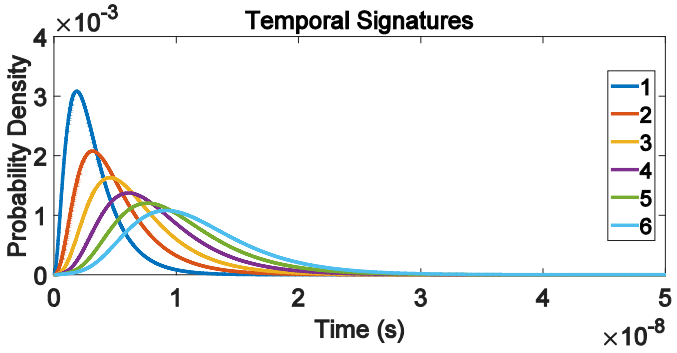

Figure 12: The temporal signatures of the six RET networks, i.e., the conditional PDF of the time to fluorescence from the chromophore AF750 in each of the six RET networks. The index of each RET network equals the number of mediators (AF594's) it contains.

As shown in Figure 12, the mean time to fluorescence from the AF750 chromophore increases with the number of mediators, which can be intuitively explained by a longer time an average exciton takes to reach the AF750 in a longer chromophore wire. Additionally, because each RET transfer incurs a convolution with an exponential distribution, the distribution of the time to reach the AF750 becomes less concentrated in a longer wire and resembles a hypoexponential distribution with more exponential stages. It should be noted that they are not exactly hypoexponential distributions due to the nonzero back transfer rates. The difference between two temporal signatures determines the difficulty of discriminating them, and can be quantified as the Kullback–Leibler (KL) divergence between two phase-type distributions. For the six temporal signatures in Figure 12, their pairwise KL divergences are shown in Table 1. Given a detection resolution and signal strength, the difference

between any two signatures should exceed a threshold to maintain a low misidentification probability, which will be further discussed in the next section.

Table 1: The pairwise KL Divergences of the six temporal signatures in Figure 12.

	1	**2**	**3**	**4**	**5**	**6**
1	0	0.2649	0.8117	1.4734	2.1374	2.7151
2	0.2725	0	0.1439	0.4851	0.9303	1.4033
3	0.8045	0.1361	0	0.0974	0.3417	0.6719
4	1.4351	0.4385	0.0922	0	0.0728	0.2600
5	2.1124	0.8240	0.3117	0.0698	0	0.0572
6	2.8115	1.2529	0.6046	0.2428	0.0555	0

Meanwhile, the conversion probability of a fluorescent taggant is another metric of interest in practical applications, which is the probability of the taggant emitting a fluorescence photon after excitation and proportionally affects its fluorescence intensity. For each of the six designed RET networks, the conversion probability is the probability of an exciton fluorescing from its emitter AF750 after entering the network from its antenna AF430, and, in CTMC, corresponds to the absorption probability of the absorbing state for the emitter fluorescence $P_{A\_ef} = Prob\left(X\left(\infty\right) = S_{A\_ef}\right)$. The conversion probabilities of the six chromophore wires from short to long are respectively 0.0301, 0.0127, 0.0051, 0.0021, 0.0008, and 0.0003, the decrease of which indicates a higher probability of an exciton decaying in the middle of a longer chromophore wire. While chromophores can be placed closer in RET networks to increase their conversion

probabilities, the divergence between their temporal signatures may decrease and make taggant identification more challenging, which poses a tradeoff in the design process. Nevertheless, if uniform fluorescence intensity is required from all taggants, parameters such as the concentration of each fabricated RET network ensemble can be increased to compensate for their different conversion probabilities.

4.2 Detection method

To identify fluorescent taggants with spectrally coded signatures, their absorption or emission spectra need to be fully or partially characterized and analyzed using statistical methods such as principal component analysis (PCA) and cluster analysis [68-71]. Therefore, a spectrometer or spectral filters are required to select different wavelengths for each detection procedure. Moreover, a time-gated measurement is often necessary to minimize the effect of background emission and other noise sources [68, 72].

In contrast, a time-resolved photon detection system with a single pair of interrogation and detection wavelengths facilitates the detection of all taggants when their fluorescent signatures are encoded in the time domain. The process of single photon arrivals in TCSPC is captured in a multinomial distribution model and the taggant identification method is based on Maximum Likelihood Estimation which yields a misclassification probability that exponentially decreases with the number of detected photons. The number of required photons to guarantee a high identification accuracy

can be as low as a few hundred, which is orders of magnitude lower than in spectrally-coded approaches. Further, the identification of a mixture of taggants, in multiplex detection, can also be formulated in MLE and reliably resolved by the Expectation Maximization (EM) algorithm [73].

4.2.1 Detection system

A prototype detection system for the temporally coded fluorescent taggants is built (see Figure 13). Powered by a pulsed laser diode driver, a laser diode emits ultrashort pulses to interrogate the RET network based fluorescent taggant in a sample cuvette. A spectral filter can be inserted between the laser diode and the sample cuvette if the laser spectrum needs to be attenuated. The fluorescence of the taggants are focused onto the aperture of a single-photon avalanche diode (SPAD), and a spectral filter is placed before the SPAD to only pass the wavelength region where temporal signatures are encoded. With a timing resolution as high as ~40ps, the photon detection signal of the SPAD is fed into a Time-Correlated Single Photon Counting (TCSPC) module [74]. This time-resolved photon counting module records the detection times of individual photons relative to the SYNC signal with a timing resolution as high as several picoseconds and reconstructs the time-resolved histogram of photon counts. The TCSPC measurements of fluorescent taggants are usually performed in a laboratory where steps are taken to minimize the background noise. Portable TCSPC systems have recently been built and in situ detection has been demonstrated [75, 76]. When a lower timing

resolution is sufficient for distinguishing temporal signatures, a high-speed gated ICCD camera may be used instead of a TCSPC.

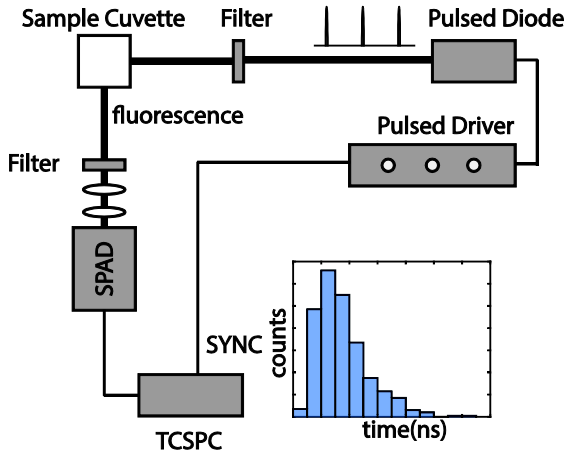

Figure 13: A prototype detection system for the fluorescent taggants with temporally coded signatures in a lab demo.

4.2.2 Taggant identification based on Maximum Likelihood Estimation (MLE)

Because RET networks are entropy sources with different temporal probability distributions and the photon detection times are the random numbers generated from one of these distributions, the process of taggant identification is to estimate the unknown generating distribution behind these observed random numbers. The process of single photon counting is captured by a multinomial distribution in TCSPC; thus MLE can be applied to reliably and conveniently identify a taggant. In the special case of single exponential decays, the methods of estimating fluorescence lifetime have been extensively investigated, and MLE is the most statistically efficient estimator and asymptotically achieves the Cramér–Rao lower bound (CRLB), which is the theoretical

limit of the variance of any unbiased estimator [77, 78]. Only a few hundred photons are needed to yield an estimate of lifetime with 10% relative error when the background photon count is negligible, and the required number of photons only slightly increases in the presence of 20% background fluorescence [78]. Additionally, the Kullback-Leibler minimum discrimination information has been successfully used to classify a measured signal among a set of lifetimes in low light conditions [79, 80]. Equivalent to MLE in a finite regime, this method has the lowest misclassification probability.

Beyond single exponential decays, the Kullback-Leibler minimum discrimination information can also be used to locate the temporal taggant signature that best matches a measured signal because this classification method only requires the multinomial distribution model of photon detections. Assuming a taggant library contains S taggants, the temporal signature of each taggant is represented by the probability density function (PDF) of a temporal probability distribution $f_i(t)(i=1,\ldots,S)$. The detection system has a finite measurement window T, which is equally divided into k channels of width ΔT. For taggant i, the probability of a photon being detected in channel j is $p_j(i)$ in the absence of background photons (Eq. (9)).

$$p_j(i)=\frac{\int_{(j-1)\Delta T}^{j\Delta T} f_i(t)\,dt}{\int_0^T f_i(t)\,dt}\left(j=1,\ldots,k\right) \qquad (9)\text{a}$$

$$\sum_{j=1}^{k} p_j(i) = 1 \tag{9)b}$$

If N photons are detected from taggant i, their distribution over the k channels follows a multinomial distribution (Eq. (10)), where $n = [n_1, \ldots, n_k]$ is the number of photons in each channel and $\sum_{j=1}^{k} n_j = N$. Given sufficient measurement window T and number of channels k, each type of taggant has a unique pattern of distributing photons, which is a multinomial distribution parameterized by $[p_1(i), \ldots, p_k(i)](i = 1, \ldots, S)$.

$$P(n \mid Taggant = i) = \frac{N!}{n_1! \ldots n_k!} p_1(i)^{n_1} * \ldots * p_k(i)^{n_k} \tag{10}$$

Within this context, the task of taggant identification becomes to classify the measured signal, i.e., $n = [n_1, \ldots, n_k]$, among the S patterns. The Kullback-Leibler minimum discrimination information is calculated between the measurement and each pattern (Eq. (11)). With the calculated minimum discrimination information for all S patterns, the detected taggant is expected to be the one that yields the lowest value of I^*.

$$I^*(i) = \sum_{j=1}^{k} n_j * ln\left(\frac{n_j}{N p_j(i)}\right)(i = 1, \ldots, S) \tag{11}$$

When classifying a measured fluorescence signal using the Kullback-Leibler minimum discrimination information, the probability of misclassification has been

theoretically derived and experimentally verified for dyes with single exponential decays [80]. Because the results are based upon the multinomial distribution model, they are applicable to general temporal signatures. The probability of incorrectly classifying a detected taggant of type i as type i' ($P(i'|i)$) can be expressed using the Gaussian error function by approximating $\Delta I_{i'i} = I^*(i') - I^*(i)$ as a normal distribution (Eq. (12)). This misclassification probability decreases exponentially with the signal strength N.

$$P(i'|i) = \frac{1}{2} - \frac{1}{2}\mathrm{erf}\left(\sqrt{\frac{N}{2v_{i'i}}}e_{i'i}\right) \tag{12}a$$

$$e_{i'i} = \sum_{j=1}^{k} p_j(i) \cdot ln\left(\frac{p_j(i)}{p_j(i')}\right) \tag{12}b$$

$$v_{i'i} = \left\{\sum_{j=1}^{k} p_j(i) \cdot \left[ln\left(\frac{p_j(i)}{p_j(i')}\right)\right]^2\right\} - e_{i'i}{}^2 \tag{12}c$$

When more than two taggants exist in the taggant library ($S > 2$), the error probability of misclassifying taggant i is

$$P\left(\bigcup_{i' \neq i} i'|i\right) \leq \sum_{i' \neq i} P(i'|i). \tag{13}$$

Consider the six temporal signatures in Figure 11 and a measurement window $T = 50ns$ with $k = 256$ equally divided channels. The error probability of misclassifying each taggant predicted by Eq. (13) is plotted in Figure 14. Taggants 5 and 6 have higher misclassification probabilities than the others due to the shorter KL divergence between their temporal signatures. Nevertheless, the misclassification probability of each taggant

decreases exponentially with the number of detected photons, and only 500 photons are needed to reach the accuracy of at most 1 error in 10,000 classifications for all taggants.

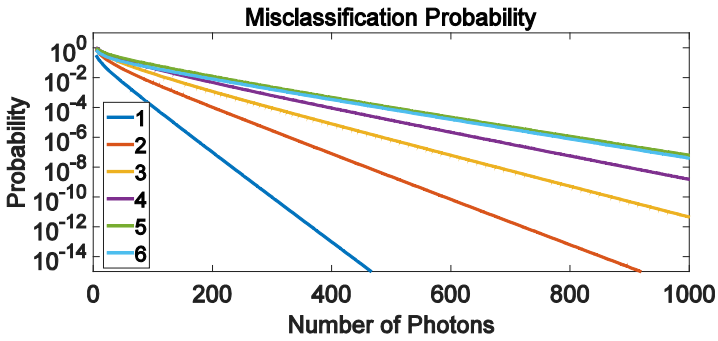

Figure 14: The error probability of incorrectly classifying each taggant among the six fluorescent taggants in Figure 11.

If the taggant detection is carried out in a low-light environment and the background emission is filtered out by the spectral filter before the SPAD, it is reasonable to neglect background photons because of the extremely low dark count rate (~2Hz) of modern SPAD's [74]. However, when they are not negligible, the background photons can be modeled as a uniform distribution over the k channels in the measurement window [78]. For taggant i, the probability of detecting a photon in channel j in the presence of background photons is now

$$p'_j(i) = \frac{b}{k} + (1-b)\,p_j(i) \quad (j = 1,\ldots,k), \tag{14}$$

where $p_j(i)$ is the probability of a fluorescence photon being detected in channel j in a background-free environment (Eq. (9)) and b is the portion of photons due to the background noise. When the modified patterns of detecting photons ($p'_j(i)$) of the six

taggants are used for taggant identification in a background noise of b, the number of photons required for the same identification accuracy of 1 error in 10,000 classifications increases with b (see Figure 15). The increase remains trivial when $b < 30\%$, and fewer than 1,000 photons are needed in this region. Aside from this theoretical analysis of the effect of noise on the identification accuracy, a practical approach to incorporating noise and other non-ideal aspects (e.g., the instrument response function of the detection system) is to accurately measure the pattern $p_j(i)$ of each taggant in a similar or identical environment prior to taggant detections and use the measured patterns as the reference for classification.

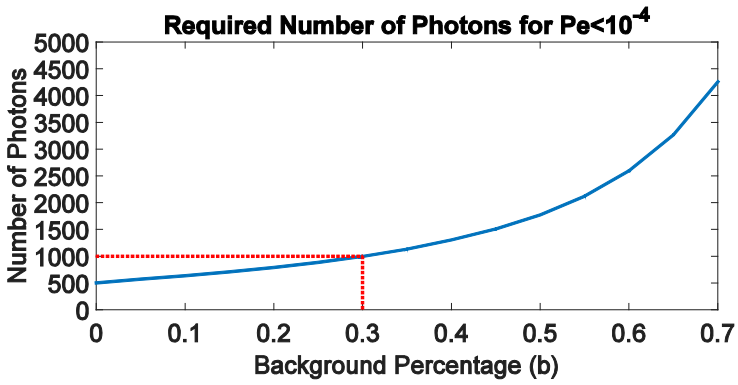

Figure 15: The required number of photons increases with the percentage of background noise (b) to keep the misclassification probability of the six fluorescent taggants below 0.0001.

4.2.3 Multiplex detection

Multiple taggants sometimes exist as a mixture, and it is often desired to recognize each constituent taggant. Least squares based methods are commonly used to resolve a mixture of fluorescent taggants with spectrally coded signatures in multiplex

detection [81-83]. However, they often require an even finer spectral characterization due to the assumed Gaussian noise model and produce ambiguous results as the number of taggants or their spectral overlap increases.

In contrast, a mixture of temporally coded signatures can be more reliably and conveniently resolved using the statistical methods for mixture models such as the EM algorithm. Assume N photons have been detected from a sample and their distribution over the k channels is $n=\left[n_1,\ldots,n_k\right]$. Given the prior knowledge that the sample is a mixture of two taggants from the six taggants in Figure 10, their identities (Tag_1 and Tag_2) and fractions (p and $1-p$) constitute a mixture model behind the measured time-resolved signal. The photon distribution given this mixture model follows a multinomial distribution:

$$P\left(n \mid p\cdot Tag_1+(1-p)\cdot Tag_2\right)=\frac{N!}{n_1!\ldots n_k!}p_1^{'n_1}*\ldots*p_k^{'n_k}; \quad (15)\text{a}$$

$$p_j^{'}=p\cdot p_j\left(Tag_1\right)+(1-p)\cdot p_j\left(Tag_2\right)\left(j=1,\ldots,k\right), \quad (15)\text{b}$$

where $p_j\left(Tag\right)\left(j=1,\ldots,k\right)$ is the probability of detecting a photon in each time bin given a specific taggant (Eq. (9)). The parameters of this mixture model can be estimated through maximizing the likelihood of observing the measured signal, i.e., $\sum_{j=1}^{k}n_j\cdot\log\left[p\cdot p_j\left(Tag_1\right)+(1-p)\cdot p_j\left(Tag_2\right)\right]$. However, this likelihood is not convenient to directly optimize due to the sum inside the logarithm. Instead, the EM algorithm dynamically calculates the probabilities of a detected photon being from the two

taggants, i.e., $T_{j,1}$ and $T_{j,2}\,(j=1,\ldots,k)$, and optimizes the target likelihood through iteratively evaluating and maximizing the expected log-likelihood $\sum_{j=1}^{k} n_j \left\{ T_{j,1} \log\left[p \cdot p_j\left(Tag_1\right)\right] + T_{j,2} \log\left[\left(1-p\right)\cdot p_j\left(Tag_2\right)\right]\right\}$. As a result, the parameters can be separately updated in each iteration until the target likelihood reaches its maximum, which is outlined as follows.

1. $Tag_1 = 1; Tag_2 = 6; p = 0.5.$ *// starting point of the three variables*
2.
   2.1 E-step:
   With the current estimates of Tag_1, Tag_2, *and* p, *calculate* $T_{j,1}$ *and* $T_{j,2}\,(j=1,\ldots,k)$*:*
   $$T_{j,1} = \frac{p \cdot p_j\left(Tag_1\right)}{p \cdot p_j\left(Tag_1\right) + \left(1-p\right)\cdot p_j\left(Tag_2\right)};\ T_{j,2} = \frac{\left(1-p\right)\cdot p_j\left(Tag_2\right)}{p \cdot p_j\left(Tag_1\right) + \left(1-p\right)\cdot p_j\left(Tag_2\right)}.$$
   2.2 M-step:
   With the latest values of $T_{j,1}$ *and* $T_{j,2}\,(j=1,\ldots,k)$, *update* Tag_1, Tag_2 *and* p*:*
   $$p = \sum_{j=1}^{k} n_j T_{j,1} / N;$$
   $$Tag_1 = \underset{Tag_1}{arg\,max} \sum_{j=1}^{k} n_j T_{j,1} \log p_j\left(Tag_1\right);$$
   $$Tag_2 = \underset{Tag_2}{arg\,max} \sum_{j=1}^{k} n_j T_{j,2} \log p_j\left(Tag_2\right).$$
   2.3 if termination condition is met:
   break
   else:
   go back to 2.1
3. *end*

After the three parameters are initialized with a starting point, the algorithm enters the iterations of Expectation and Maximization steps. Each iteration consists of an Expectation step and a Maximization step. The Expectation step calculates the probabilities of a detected photon being from the two taggants, i.e., $T_{j,1}$ and

$T_{j,2}\left(j=1,\ldots,k\right)$, based on the current estimates of the identities and fractions of the two taggants. With the current values of $T_{j,1}$ and $T_{j,2}$ as the weights of each detected photon related to the two taggants, the Maximization step updates the fractions of the two taggants and their identities by respectively maximizing the weighted likelihood of observing the photon detection times for each taggant. The iterative process reaches an end when a termination condition is met. For example, the target likelihood can be evaluated at the end of each iteration, and when the change of this value between adjacent iterations falls below a threshold, it can be concluded that the target likelihood has been maximized. Because, dependent on its starting point, the EM algorithm may converge to a local maximum, it may be necessary to run the EM algorithm with different starting points to improve the chance of locating the maximum likelihood and the correct values of the three parameters.

Because this approach to multiplex detection is still based on the multinomial distribution model and MLE, it is the most statistically efficient. Further, a model selection criterion such as Bayesian Information Criterion (BIC) can be used to estimate the number of existing taggants in a mixture if this information is absent prior to a detection. However, it is noteworthy that a higher number of photons is often necessary in multiplex detection because more parameters are to be estimated in the mixture model.

4.3 Lidar integration

Lidar is a remote sensing technology that identifies objects of interest and measures distance by illuminating a target with a laser and analyzes the reflected light, which has wide applications in archaeology, forestry, atmospheric physics, anti-poaching, etc. Because the fluorescence emitted by a target is often used to determine the presence of the target and its distance, natural or artificial fluorescent materials become common targets in Lidar applications [68-72, 84]. However, these fluorescence Lidar applications usually use the spectral characteristics of the fluorescent materials by measuring the intensity of the reflected light within one or multiple wavelength bands, which has limited the ability to tag different objects and resolve them in multiplex detection.

With the unique advantages of a RET network based fluorescent taggant with temporal signatures, a large number of different objects can be conveniently coded in the same wavelength region with a small set of chromophores. The single channel of interrogation and detection wavelengths will make the detection procedure highly efficient, and the superior reliability of target identification under low light conditions will potentially increase the detection range.

While the fluorescence being detected in fluorescence Lidar applications is often in the wavelength range between 350nm and 710nm which overlaps the solar spectrum, its spectral and temporal characteristics can still be accurately measured by taking

measures to reduce the background signal such as using optical filters and time-gating techniques. Modern ICCD cameras are highly sensitive cameras that have single photon detection capability and high speed time-gating (min. gate width ~5ns) capability with a resolution of ~40ps. Many modern fluorescence Lidar systems use ICCD cameras and have high-resolution temporal measurement capability, and they can operate in full daylight [85-87]. These implementations are compatible with the temporally coded fluorescent taggants if their timing resolution is sufficient to resolve the temporal signatures. Combined with the high-speed time-gating technique, TCSPC can potentially further improve the timing resolution when it becomes necessary for taggant discrimination.

There are additional aspects to take into account when designing a fluorescent taggant for Lidar applications, which are partially considered in the previous work [72] that fabricated fluorescent taggants using nanocrystals and demonstrated their far-field (~3km) detection capability. For example, the laser source in common fluorescence Lidar applications often has a UV wavelength around 350nm due to the higher eye-safe power density in this region, and commercially available dyes suitable for this excitation wavelength should be considered as the source chromophore such as ATTO 390 and Alexa Fluor 350. In addition, the mediator and the emitter should have low absorptivity so that their direct excitation is reduced and their excitation mostly comes from the source chromophore through RET transfer.

4.4 Summary

As an entropy source that can be programmed to generate random samples from different distributions, RET networks are used to make fluorescent taggants with temporally coded signatures. Compared with spectrally or lifetime coded fluorescent taggants, these temporally coded fluorescent taggants have a significantly larger coding capacity and flexibility in taggant design. Meanwhile, the taggant detection and identification process becomes highly efficient and reliable even with only a few hundred photons under low light conditions, and can also resolve a mixture of taggants in multiplex detection. These unique properties make the temporally coded fluorescent taggants a superior candidate for both *in situ* and Lidar applications.

While this application mainly leverages the flexible programmability of the probability distributions of RET networks to create a large taggant library, the next application will demonstrate how the efficient random number generation of RET networks can be used to accelerate probabilistic algorithms.

5. Application in Stochastic Computing

Statistical methods, machine learning in particular, are increasingly used to address important problems including, but not limited to: computer vision, robot/drone control, data mining, global health, computational biology, and economics. Many approaches in statistics and machine learning utilize probabilistic algorithms that generate samples from parameterized probability distributions. Probabilistic algorithms are the only viable approach to the exact solution of many classes of important problems (e.g., high-dimensional inference, rare event simulation). Meanwhile, they are more suitable than deterministic algorithms for general applications when the goal is to seek the most probable outcome rather than its accurate probability [88], and offer the potential to create generalized frameworks with close ties to reality [89].

Despite the theoretical advances in statistics and probabilistic machine learning, the fundamental mismatch persists between the deterministic hardware that traditional computers use and the stochastic nature of probabilistic algorithms. Modern computers largely take a deterministic approach to computation and are designed with deterministic algorithms and transistor functionality in mind. Recent challenges in CMOS scaling reveal practical limits on performance as lithographic features continue to shrink.

Within this context, the challenge we propose is to develop new hardware that directly supports a wide variety of probabilistic algorithms [26]. Based on the

observation that novel probabilistic functional units can be created using RET networks to approximate arbitrary probabilistic behavior and generate random samples from general distributions [18], this dissertation takes the first steps toward meeting this challenge by exploiting the physical properties of the molecular-scale photonic device.

To meet the above challenge, we introduce the concept of a RET-based Sampling Unit (RSU), a hybrid CMOS/RET functional unit that generates samples from parameterized distributions. An RSU specializes the calculation of distribution parameters in CMOS and uses RET to generate samples from a parameterized distribution in only a few nanoseconds. There are a variety of distributions that could be implemented by an RSU; however, in this work we focus on an RSU that implements a distribution for use in a particular class of Bayesian Inference problems.

Bayesian Inference is an important, generalized framework that estimates a hypothesis (i.e., values for random variables) using a combination of new evidence (observations) and prior knowledge. Markov Chain Monte Carlo (MCMC) sampling is a theoretically important and powerful technique for solving the inference problem that iteratively and strategically samples the random variables and ultimately converges to an exact result. However, MCMC becomes inefficient for many inference problems in practice, especially those with high dimensionality (many random variables) and complex structure. MCMC can require many iterations to converge to a solution and the inner loop incurs the overhead of sample generation from prescribed distributions.

Although deterministic inference algorithms can be faster than MCMC, they sacrifice accuracy (e.g., by approximation) and require more complex mathematical derivation. Similarly, problem specific non-Bayesian algorithms forego the benefit of a generalized framework and require reformulation for each problem.

Markov Random Field (MRF) Bayesian Inference can be used for a broad class of applications, including image processing (e.g., image segmentation, motion estimation, stereo vision, texture modeling), associative memory, etc. The overall goal is often to determine the most likely value for each random variable given the observed data (i.e., marginal MAP estimates). Given a specified MRF model, this is achieved in MCMC by iteratively sampling the random variables according to the conditional dependencies and then identifying the mode of the generated samples.

To accelerate MRF inference using MCMC, we introduce RSU-G, a Gibbs Sampling unit based on the 'first-to-fire' exponential sampling units described in Section 3.3.2.3. Our specific RSU-G unit supports first-order MRFs with a smoothness-based prior, which includes many image processing applications (e.g., image segmentation, motion estimation, stereo vision). A survey of possible applications is provided elsewhere [90].

The proposed molecular-scale optical Gibbs sampling unit (RSU-G) can be integrated into a processor / GPU as specialized functional units or organized as a discrete accelerator. Emulation-based evaluation of two computer vision applications for

HD images reveal that an RSU augmented GPU provides speedups over a GPU of 3 and 16. Analytic evaluation shows a discrete accelerator that is limited by 336 GB/s DRAM produces speedups of 21 and 54 versus the GPU implementations. The novel optical components of RSU-G units consume very little power (0.16 mW) and area (0.0016 mm^2). Synthesizing the CMOS portions of RSU-G in 15nm reveals power of 3.75 mW and area of 0.0013 mm^2 for a total RSU-G power of 3.91 mW and area of 0.0029 mm^2.

5.1 Sampling and probabilistic computing

The increasing use of machine learning and probabilistic algorithms in data analytics presents new challenges and opportunities for the design of computing systems. This section provides a brief overview of probabilistic algorithms, their challenge in practice due to sampling overhead, a potential solution based on RET circuits, and presents related work.

5.1.1 Probabilistic algorithms

Recent theoretical advances in statistics and probabilistic machine learning demonstrate that many application domains can benefit from probabilistic algorithms in terms of simplicity and performance [91, 92]. Example problems include, but are not limited to, statistical inference, rare event simulation, stochastic neural networks (e.g., Boltzmann machines), probabilistic cellular automata and hyper-encryption.

Most probabilistic computations rely on sampling from application-specific distributions. For example, Bayesian Inference solvers often iteratively sample from

common distributions such as gamma distribution and normal distribution, while rare event simulations may require many samples from a heavy-tailed distribution to obtain statistically significant results. The number of samples generated in these applications can be large; many thousands of samples per random variable with thousands of random variables.

The importance of sampling from various distributions led to the C++11 standard library including implementations for 20 different distributions. Although this greatly simplifies program development, it does not address the inherent mismatch between conventional digital computers and probabilistic algorithms. In particular, sampling requires control over a parameterizable source of entropy used for random selection. Therefore, generating a sample includes two critical steps: 1) parameterizing a distribution and 2) sampling from the distribution.

Consider Bayesian Inference, an important inference framework that combines new evidence and prior beliefs to update the probability estimate for a hypothesis. Consider D as the observed data and X as the latent random variable. $p(X)$ is the prior distribution of X, and $p(D|X)$ is the probability of observing D given a certain value of X. In Bayesian Inference, the goal is to retrieve the posterior distribution $p(X|D)$ of the random variable X when D is observed. As the dimension of $X = [X_1, \ldots, X_n]$ increases, it often becomes difficult or intractable to numerically derive the exact posterior distribution $p(X|D)$. One approach to solve these inference problems uses

probabilistic Markov Chain Monte Carlo (MCMC) methods that converge to an exact solution by iteratively generating samples for random variables. Obtaining each sample incurs at least the overhead of computing the distribution parameters and sampling from the distribution.

5.1.2 Sampling overhead

Parameterizing a distribution is application dependent and requires computing specific values for a given distribution. For example, computing the decay rate for an exponential, the mean and variance for a normal, etc. For a class of Bayesian Inference problems that we study this may include computing a sum of distance values and can take at least 100 cycles to compute on an Intel E5-2640 processor (compiled with gcc –O3), and could be much higher. Other probabilistic algorithms may have different computations with varying complexity. Nonetheless, the time required to parameterize a distribution is an important source of overhead in probabilistic algorithms.

The second component for sampling is to generate the sample from the parameterized distribution. Devroye [93] provides a comprehensive overview of techniques for computationally generating samples from various distributions. Samples from general continuous or discrete distributions can be generated using algorithms such as inverse transform sampling and rejection sampling. Unfortunately, it can take hundreds of cycles to generate a sample with these approaches, and more complex multivariate sampling can take over 10,000 cycles. Table 2 shows how many cycles it

takes to generate a sample for a few distributions using the C++11 library [94]. We obtain cycle counts on an Intel E5-2640 using the Intel Performance Counter Monitor and present the average of 10,000 samples (-O3 optimization).

Table 2: Cycles to sample from different distributions.

Distribution Type	Cycles (average)
Exponential	588
Normal	633
Gamma	800

The overheads of calculating distribution parameters and sampling from the distribution are critically important to many probabilistic algorithms since they incur both overheads in their inner loop. Furthermore, multiple applications often use the same distribution and share the computation to parameterize the distribution. Therefore, accelerating sampling can have a significant impact on overall execution time.

5.1.3 RET Circuit

Nanoscale physical samplers can be built using RET networks to efficiently and directly generate samples from general distributions. As described in Chapter 3, RET networks can approximate virtually arbitrary probabilistic behavior since they can implement sampling from phase-type distributions, and the probabilistic behavior of a RET network is determined by its physical geometry. Typically, a given RET network corresponds to a specific distribution. Sampling from the distribution occurs by

illuminating the RET network and detecting output fluorescence photons as a function of time (within a few nanoseconds).

Therefore, RET networks are integrated with an on-chip light source, e.g., quantum-dot LEDs (QD-LEDs), waveguide, and single photon avalanche detector (SPAD) to create a RET circuit (Figure 16). Each RET circuit can contain an ensemble of RET networks. A fully specified RET network can be conveniently and economically fabricated with sub-nanometer precision using hierarchical DNA self-assembly [31, 32]. RET circuits can be integrated with hybrid electro-optical CMOS using back end of line processing [95, 96]. Because samples are directly generated from the distribution of a RET network through single photon detection, the sample generation process only takes a few nanoseconds for any distribution. The RET-based sampler may also reduce power consumption since the samples are generated in the form of single fluorescent photons.

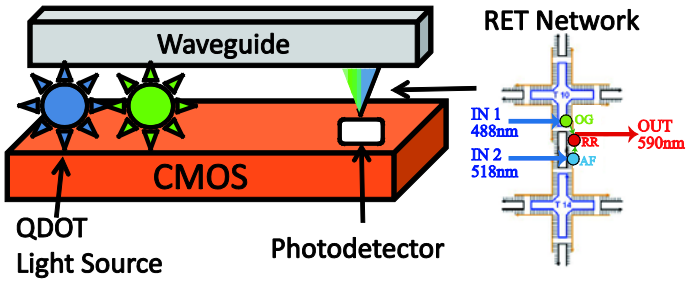

Figure 16: A RET circuit.

Chapter 3 outlined different implementations of continuous and discrete samplers. In this dissertation, we focus on discrete samplers for Markov Chain Monte

Carlo (MCMC) solvers for Markov Random Field (MRF) Bayesian Inference that utilize RET-based exponential samplers (Section 3.3.2.3).

To accelerate probabilistic algorithms by reducing its sampling overhead, we create RET-based Sampling Units (RSU), which are probabilistic functional units that use CMOS specialization to accelerate distribution parameterization and RET circuits to accelerate obtaining a sample from the parameterized distribution. Section 5.2 provides an overview of a generic RSU, and Sections 5.3 and 5.4 describe the details of a specific RET-based Gibbs Sampling Unit (RSU-G) to accelerate MRF inference using MCMC, and Section 5.5 discusses the processor architectures for using these units and their evaluation. The remainder of this section briefly discusses alternative approaches.

5.1.4 Alternative approaches and related work

Reducing or avoiding the overhead of sampling can be achieved by using deterministic algorithms or by introducing hardware specialization. One approach to avoid the overhead of sampling is to use alternative discrete algorithms. For example, an alternative to MCMC is deterministic approximate inference methods such as Expectation Propagation (EP) and Variational Bayesian (VB). Although often more efficient in practice, these methods require more complex mathematical derivation and arbitrary assumptions that create divergence from the exact solution [92, 97, 98], whereas only sampling algorithms are guaranteed to ultimately converge to the exact

solution of inference problems [99]. Domain scientists generally prefer the less complex mathematically, but more accurate pure solution if possible.

Another approach to accelerate sampling, and the one we advocate in this work, is through specialization that incorporates novel devices. A Stochastic Transition Circuit and an FPGA implementation was proposed to efficiently update random variables given the graphical model of an inference problem [63, 88]. Abstractly, our units are an instance of a Stochastic Transition Circuit; however, our approach differs in that we exploit the physical properties of RET and can implement complex distributions that would be difficult in CMOS.

Similar to an RSU, other techniques propose using a physical process as a natural source of entropy to create samplers. The common physical processes used for this fall into four categories: noise, free running oscillator, chaos and quantum phenomena [48]. With a quantum-mechanical origin [50], RET provides true randomness and arbitrary sampling distributions. While previous works on RET between quantum dots support certain probabilistic computing applications [100, 101], RSUs are more general and can be used across a broad set of applications. Although often used in CMOS to generate random bits, thermal noise cannot provide provable randomness and requires complex post-processing to make the output appear more random. Further, its implementation is either difficult to parameterize or does not support arbitrary distributions, thus limits reuse across applications.

The Intel Digital Random Number Generator (DRNG) uses a thermal noise-based entropy source and includes two stages of post-processing: an AES-based entropy conditioner and a Pseudo Random Number Generator (PRNG) [102]. Based on our synthesis of the 256-bit AES conditioner at 45nm technology [103], this stage alone is comparable to the RSU-G_1 unit proposed later in terms of area, power consumption, and throughput. The full DRNG requires more area and power.

Probabilistic CMOS (PCMOS) can implement discrete samplers by using thermal noise in electronic circuits to make probabilistic switches with a set of tunable parameters [104, 105]. Unfortunately, this requires amplifying the noise to a specific magnitude, and can be energy and area inefficient. The exact noise level of each probabilistic switch relies on the probabilities of all values and requires normalization. Furthermore, PCMOS switches essentially implement Bernoulli random variables, and cannot be flexibly organized to generate samples from general distributions.

Other recent work explores augmenting processors with specialized Neural Processing Units (NPU) [106, 107] to achieve speedup and power savings using analog circuits, but focuses specifically on using neural networks to approximate deterministic functions.

RSUs can implement a broad class of distributions, can be easily parameterized dynamically by changing RET circuit inputs (e.g., QD-LED intensity values), and eliminate normalization for some cases by using relative ratios of distribution

parameters (e.g., exponential decay rates). RSUs provide an efficient hardware platform for probabilistic programming languages [88, 108] by providing native probabilistic support. Exploring language support for RSUs is an interesting future direction, but first we must develop specific units and provide an architecture for probabilistic computing.

5.2 RET-based Sampling Unit (RSU)

To provide an efficient hardware platform for probabilistic algorithms, we introduce the concept of a RET-based Sampling Unit (RSU), a hybrid CMOS/RET functional unit that generates samples from parameterized distributions. An RSU specializes the calculation of distribution parameters in CMOS and uses RET circuits to generate samples from a parameterized distribution in only a few nanoseconds.

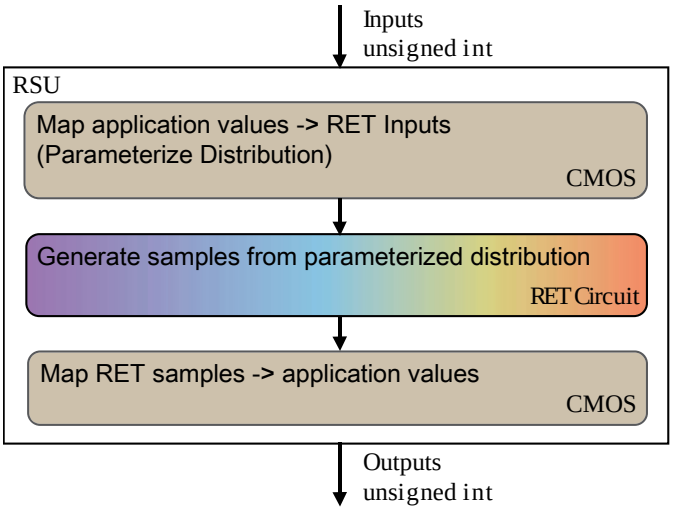

Figure 17: Generic RSU block diagram.

A generic RSU is a hybrid of CMOS and RET technology, and its inputs and outputs are unsigned integers that correspond to values of interest to the application. A

block diagram of an RSU is shown in Figure 17. An RSU performs a series of three operations: 1) map application values to RET inputs, 2) generate samples, 3) map RET output to application value. Steps 1 and 3 are implemented using conventional CMOS specialization, whereas Step 2 is a RET circuit that exploits the probabilistic behavior of RET networks. Step 1 is where distribution parameterization occurs, and involves converting application values into RET circuit inputs (e.g., QD-LED intensity values). Step 2 samples from the parameterized distribution using one or more RET circuits, and Step 3 converts the RET circuit output value back to an application data type for the sample.

RSUs can be designed for a variety of probabilistic algorithms, and exploring the large design space requires many man hours. In this work we explore a small region of the overall design space by focusing on an RSU designed specifically to accelerate a particular class of Bayesian Inference problems through MCMC, and the details of this RSU are discussed in Sections 5.3 and 5.4.

5.3 RET-based Gibbs Sampling Unit (RSU-G)

This section presents an RSU designed specifically for the MCMC solver for Markov Random Field inference problems. We first summarize Markov Random Fields and MCMC approaches to Bayesian Inference, and then present our RSU designed to accelerate this broad class of applications.

5.3.1 Markov Random Fields

A Markov Random Field (MRF) is a type of graphical model used in Bayesian Inference. An MRF is a set of random variables that satisfies the Markov properties described by an undirected graph [109]. MRFs are suitable for many interesting and important applications such as low-level computer vision and associative memory [90, 109, 110]. In this thesis, we focus on first-order MRFs with smoothness-based priors, homogeneity and isotropy (i.e., position and orientation independence), with discrete random variables. Extending our work to other types of MRFs is future work.

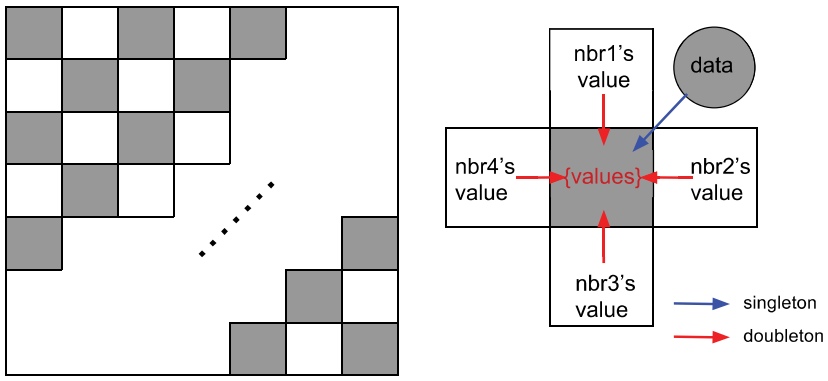

Figure 18: A first-order MRF.

Figure 18 shows an example first-order MRF where each random variable has four neighbors and is conditionally independent of all non-adjacent random variables when conditioned on the four neighbors. More specifically, the full conditional probability of each random variable $X_{i,j}$ is the exponential of the sum of five clique potential energies (with normalization):

$$p\left(X_{i,j}|X_{-(i,j)},D\right)$$
$$\propto \exp\left\{-\frac{1}{T}\left[Ec_S\left(X_{i,j},\ D\right)+Ec_D\left(X_{i,j},X_{i-1,j}\right)+Ec_D\left(X_{i,j},X_{i,j-1}\right)+Ec_D\left(X_{i,j},X_{i+1,j}\right)+Ec_D\left(X_{i,j},X_{i,j+1}\right)\right]\right\}. \quad (16)$$

$X_{i,j}$ is a random variable that can take on M possible values, or more commonly called *labels* in this context. $Ec_S\left(X_{i,j},\ D\right)$ is the singleton clique potential energy that relates $X_{i,j}$ to the observed data D, and $Ec_D\left(X_{i,j},X_{i',j'}\right)$ is the doubleton clique potential energy that relates $X_{i,j}$ to a neighboring random variable. T is a fixed constant.

5.3.2 MCMC and Gibbs sampling

For many problems, directly solving the equations above can be computationally expensive. Therefore, MCMC methods are often employed. Among the algorithms for generating MCMC samples, Gibbs sampling and Metropolis sampling are the most commonly used [92], and for our applications Gibbs sampling is applicable and used.

Gibbs sampling generates a new sample of a random variable ($X_{i,j}$) directly from its full conditional distribution when conditioned on the current labels of the other random variables. In a first-order MRF, this is achieved by calculating the probability for each of the M possible labels a random variable can take on using Eq. (16) and randomly selecting a label according to the discrete distribution.

Each MCMC iteration updates all random variables once to obtain one MCMC sample. The number of operations per iteration linearly depends on the number of possible labels for each random variable and the number of random variables. However,

different labels can be evaluated simultaneously and random variables that are conditionally independent can be updated concurrently, exposing significant parallelism for some problems. Specifically, the first-order MRF in Figure 18 allows all the gray random variables to be updated simultaneously. Similarly, all the white random variables can be updated simultaneously.

5.3.3 RSU-G design

As described in Section 3.3.2.3, a discrete sampler with M outcomes can be constructed using M exponential samplers parameterized by the probabilities of taking each outcome. Generating one new sample for $X_{i,j}$ requires M different samples, each from uniquely parameterized exponential distributions. We exploit this observation to construct a RET-based Gibbs Sampling Unit (RSU-G1) using a single RET circuit (G1).

In this paper we consider only MRFs with smoothness-based priors where the energy (i.e., logarithm of probability) of taking on a label is the sum of four doubleton clique pontential energies and one singleton clique potential energy [90]. Each doubleton clique potential energy is a measure of distance between the label being evaluated and the current label of a neighbor. Typically, the distance measure is defined for the label space and is problem specific, here we consider grayscale valued images and use a simple squared difference norm as the metric (Eq. (17)). The singleton clique potential energy is application specific; RSU-G implements it as the squared difference between two data values to directly support applications such as image segmentation, motion

estimation and stereo vision and is extendable to other applications by precomputing their singleton energy externally and sending into one data input.

$$\begin{aligned} & Ec_D\left(X_{i,j} = x, X_{i+1,j} = x'\right) \\ & = Ec_D\left(X_{i,j} = \left[x1,\ldots,xn\right], X_{i+1,j} = \left[x1',\ldots,xn'\right]\right) \\ & = d^2\left(x,x'\right) = w_D \sum_k \left(xk - xk'\right)^2 \end{aligned} \tag{17}$$

RSU-G utilizes the 'first-to-fire' design based on the property of competing exponential random variables. In this design, a RET circuit is used as an exponential sampler that generates exponentially distributed samples. The M exponential samplers are parameterized by the energies of taking on each label. The key aspect in this design is parameterizing the exponential samplers, based on the neighboring random variables' current labels and the singleton energy, and recording the time to fluorescence (TTF) from each exponential sample. The label that produces the shortest TTF is chosen as the label for $X_{i,j}$.

5.3.4 Limited precision

Previous work found that 8-bit precision for energy calculations is sufficient for generating samples from discrete distributions with up to 1,000 outcomes [63]. Given the specific distance measure we use here, only 3 bits are needed for scalar values and 6 bits for vector values in the doubleton calculation, as described later. Although extra bits can increase the number of possible labels, the energies of different labels start to overlap resulting in equal selection probability. These close or redundant labels do not

necessarily improve the solution quality; however the time required to update one random variable increases since there are more labels to be sampled. In these cases, we recommend collapsing the equally likely labels into a single label before execution.

5.4 RSU-G_1 implementation

An RSU-G implementation can take many forms. On one end of the spectrum it could be constructed to iterate over the M possible labels using a single RET circuit (RSU-G_1). On the other extreme end of the spectrum it could use M distinct RET circuits to simultaneously evaluate all M possible labels in a single step (RSU-G_M). In the middle are designs with K RET circuits that take M/K steps to obtain the result (RSU-G_k).

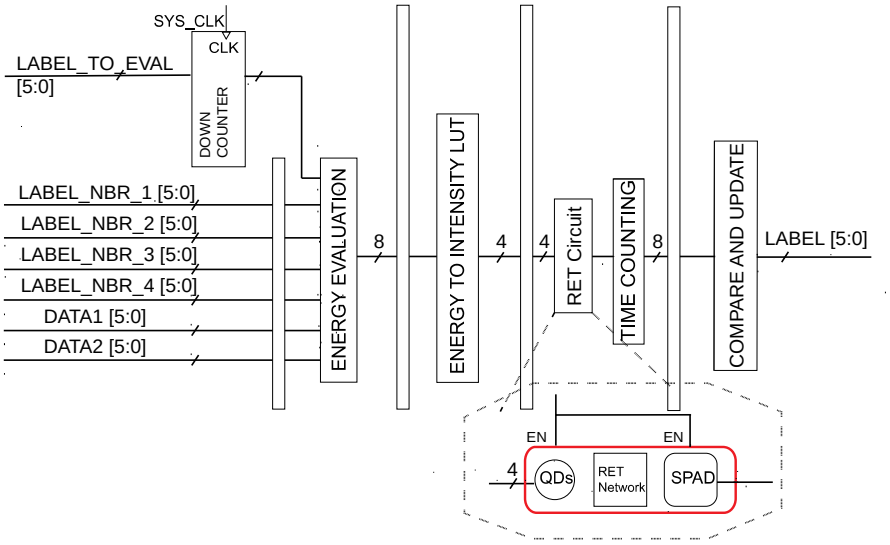

Figure 19: An RSU-G_1 implementation diagram.

5.4.1 Overview

Our preliminary RSU-G_1 implementation, shown in Figure 19, evaluates one possible label for a random variable ($K=1$) per step, and iterates to evaluate all M

labels. Given the limited label precision, we use 6-bit unsigned integers to represent random variable labels ($M<=64$). RSU-G_1 is a multicycle pipelined functional unit that takes $7+(M-1)$ cycles to obtain a random variable sample in steady state. The design can be easily extended to evaluate up to 64 labels (RSU-G_{64}) in 12 cycles at the expense of additional area. Exploring a configurabe RSU design is part of our future work.

There are five main components in any RSU-G implementation: 1) label decrement/input, 2) energy computation, 3) energy to intensity mapping, 4) RET circuits (for sampling), and 5) selection. Label decrement is used to iterate over all M possible labels. The energy computation performs the distance calculation to obtain the exponential decay rate which maps through a lookup table to a QD-LED intensity. The RET circuit samples the exponential distribution and the resulting TTF for the sample is used in the selection block to choose the lowest from all M possible labels.

For many applications M is fixed and an initial down counter value can be set at the start of the program. To obtain a sample for a random variable $X_{i,j}$, RSU-G requires five 6-bit inputs, one for each neighbor and its data value. Some applications need an additional data value that changes for each possible label. The output is a single 6-bit value that represents the new label for the random variable.

5.4.2 Pipeline stages

Here we describe each of the stages in our initial RSU-G_1 pipeline implementation. This does not necessarily represent the most optimized design.

Label. The first stage sets the inputs necessary for evaluating a given possible value for the random variable. This includes the given value to evaluate, the current values of the four neighboring random variables, and the input data values. The down counter is initialized with the maximum possible value for the random variable (M-1), and the other inputs are stored in registers. We assume this initialization is overlapped with evaluations of previous random variables. On subsequent evaluation cycles the down counter is decremented to iterate over the values, and the other values (except for the second data value) remain unchanged until the next random variable evaluation.

Energy Calculation. The second stage computes the clique potential energies, a first step in distribution parameterization. Each cycle a new energy is computed since the down counter changes. The 6-bit value can represent either a 2D vector or a scalar. For the 2D vector $[x1, x2]$, the 6-bit value is split into 3 bits for $x1$ and 3 bits for $x2$. The distance measure is calculated separately for the two entries between two neighboring random variables and then summed to obtain the doubleton energy. When the value of the random variable is a scalar, only the first entry (3 bits) is used and the second entry is set to zero. We found this limited precision to be sufficient for many applications. Similarly, the singleton energy is a distance measure between two data inputs, a calculation that is application dependent, e.g., in motion estimation it can be a weighted squared difference between two grayscale values. We assume that any scalar weights in the singleton calculation are pre-factored from the input data. The 8-bit energy for a

possible label is calculated by summing the five clique potential energies and passed onto the next stage.

Intensity Mapping. The third stage implements the second component of distribution paramaterization by mapping the 8-bit energy value to a corresponding QD-LED intensity. We use a lookup table to find the corresponding 4-bit signal that provides the input to a RET circuit to control the binary on/off state of its four QD-LEDs. The QD-LEDs are sized to provide a suitably large dynamic range of intensities to match the precision in relative probabilities we demonstrate with the RSU-G_2 hardware prototype described later.

RET Sampling. This stage activates a RET circuit to obtain a sample from the RET network. We simultaneously enable the QD-LEDs and the SPAD of the given RET circuit. The time to the first photon detection (TTF) is recorded using an 8-bit shift register that is clocked 8x faster than the system clock. It may take multiple system clock cycles before a RET circuit generates an output and can be reused for another evaluation. We elaborate on this later and use replicated RET circuits to sustain single cycle operation of the RSU-G pipeline.

Selection. In the final stage, the selection block records the shortest TTF for each possible label. Each cycle the previously shortest TTF is compared against the new TTF and the shorter is recorded as the current best TTF (with its label). After evaluating all

labels (i.e., the down counter reaches zero), the best label is returned as the new sample of the random variable $X_{i,j}$.

5.4.3 Replicated RET circuits

The TTF of a RET circuit is probabilistic, and for RSU-G follows an exponential distribution. Samples from the tail of this distribution can become arbitrarily long and the delay depends on the fluorescent lifetime of the chromophores in the RET networks. The RSU-G_1 design presented here requires four 1ns cycles for the RET circuits to reach a quiescent state, ensuring it is safe to proceed with a new sampling operation.

However, the four cycle delay creates a structural hazard in the pipeline. We use four replicated RET circuits in RSU-G_1 to overcome the hazard. This allows us to share the parameterization, timing and selection logic among all four replicates. We use a simple two-bit counter for round-robin scheduling of sampling operations across the four RET circuits and sustain a throughput of one label sample per cycle (requiring M cycles for a single random variable).

The above design represents the smallest RSU-G_1 design that produces one possible label evaluation per cycle. Utilizing more RET circuits can further reduce latency by evaluating multiple possible labels per cycle. The extreme is RSU-G_{64} that evaluates up to the maximum of 64 possible labels simultaneously by using 256 RET circuits. This design can sustain a throughput of one random variable sample per cycle. Exploring the space of RSU design is ongoing work.

5.5 RSU architectures

There are many possible ways to expose an RSU to software, ranging from adding functional units to an existing processor to a discrete accelerator. The goal in this section is to present two potential architectures that utilize RSUs: 1) augmenting a GPU with sampling units and 2) a discrete accelerator designed to maximize memory bandwidth utilization, and the preliminary evaluation of these systems in terms of performance, power and area.

5.5.1 Potential architectures

The operation of the RSU-G unit can be viewed in terms of operations performed once: 1) per application, 2) per MCMC iteration, 3) per random variable evaluated (a pixel in our applications), and 4) per potential random variable label. Each RSU-G unit requires initializing the intensity map table and down counter (max label value) at the start of each application. For each random variable (pixel in our applications), RSU-G requires the four neighbor labels (for doubleton calculations) and its associated data (e.g., grayscale value for singleton calculations). Finally, for each potential label, the singleton calculation may also need information from a target location (pixel grayscale).

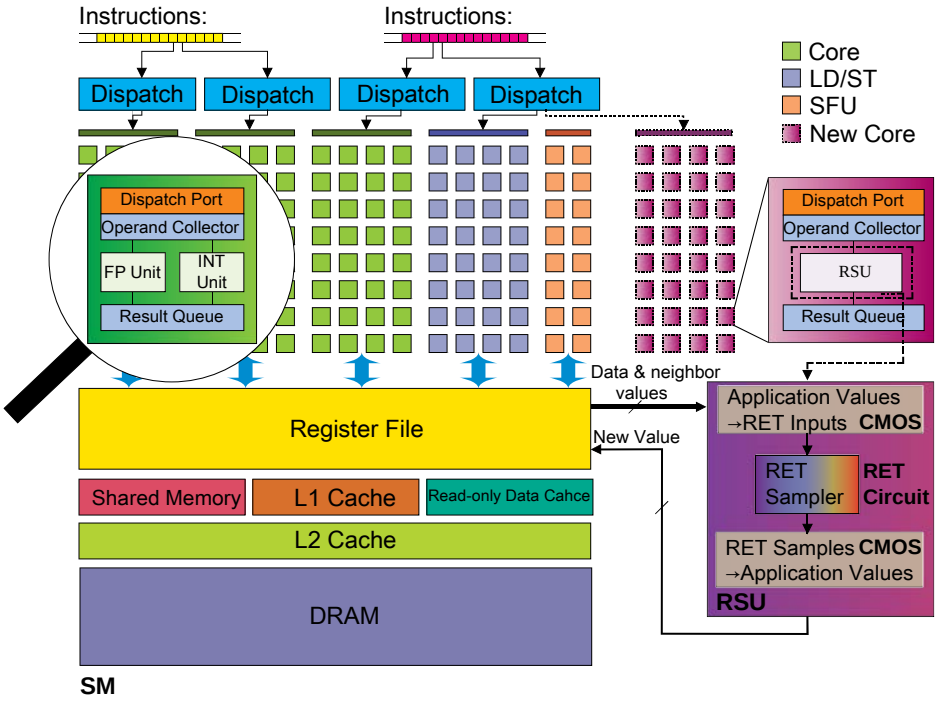

Figure 20: GPU augmented with RSUs.

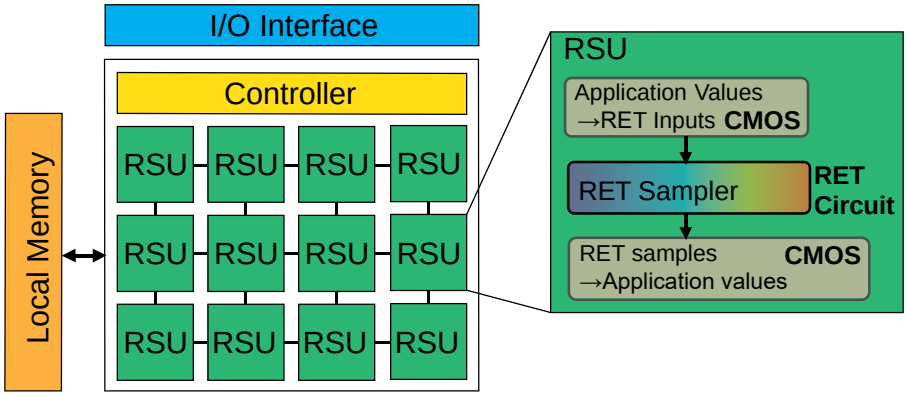

Figure 21: A discrete accelerator using RSUs.

Figure 20 shows a GPU architecture (single streaming processor) modified to include RSUs and Figure 21 shows a custom discrete accelerator design using RSUs. Augmenting a conventional CPU would be similar to the GPU design. For processor integration, additional instructions are required to access the RSU, the details of which

are discussed in our paper [111]. From a program's perspective, an RSU is a multi-cycle, pipelined functional unit that takes several random variable values as input and produces a single random variable value as output. The values are represented as unsigned integers.

An alternative to augmenting an existing processor or GPU with RSU-G units is to design a custom discrete accelerator. This removes the constraints placed on general-purpose cores to support a wide variety of applications and instead allows us to focus on achieving the maximum performance. This design assumes that all control and data movement is implemented using custom logic where datapaths and register sizes can be specialized to match RSU-G's. We expect this to be the highest performing approach and analytically investigate this upper limit on performance.

5.5.2 Evaluation

We use several image-processing applications (image segmentation, stereo vision matching, and dense motion estimation) to evaluate the RSU based systems in terms of performance, power and area. Our image segmentation application assigns one of five possible values (labels) to each pixel by grouping similar pixels based on intensity [112, 113]. The stereo vision application similarly assigns one of 5 labels to align two images [114]. The dense motion estimation application searches over a 7x7 block to find the most likely position of a pixel in a subsequent frame (49 possible values) [115]. Although application specific implementations for these problems exist, our goal is to demonstrate

the potential of RSU-G for the general MRF-MCMC Bayesian Inference framework. We use a single core of an Intel E5-2640 for image segmentation and stereo vision, but focus primarily on using an NVIDIA GTX Titan X GPU, for image segmentation and motion estimation.

5.5.2.1 Performance

We use a combination of emulation and analytic evaluation to obtain performance estimates of architectures that incorporate RSU-Gs [111]. Figure 22 illustrates the speedups for our two applications with different image sizes on a GPU augmented with RSUs. For image segmentation, RSU-G_1 systems provide speedup of 3.2 over the baseline GPU for images of size 320x320, and 3.0 for HD images of size 1080x1920. Speedups over the optimized GPU implementation are 2.5 and 2.4 for small and HD images, respectively. For dense motion estimation, RSU-G_1 systems provide speedup of 6.4 and 12.8 for 320x320 images and achieve 7.5 and 16.06 for HD images. Dense motion estimation benefits from a wider RSU-G_4 design since it has more labels to evaluate (M=49) and achieves speedups of 23 for small images and 34 for HD images over the baseline GPU implementation.

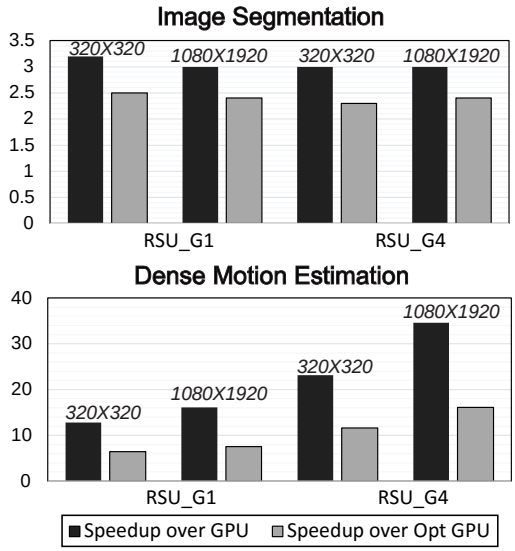

Figure 22: RSU Speedup over GPU.

Discrete Accelerator. Assuming DRAM bandwidth is 336GB/s, the GTX Titan X bandwidth, and the accelerator consumes data at DRAM bandwidth. For image segmentation, the accelerator achieves an additional 12.1x and 7x speedup over the RSU-G_1 augmented GPU for 320x320 and 1080x1920 (HD) images, respectively. Dense motion estimation achieves additional speedup of 6.5x and 3.4x for 320x320 and 1080x1920 images, respectively. The lower speedup for HD images is because HD images saturate the GPU while 320x320 images don't. Thus, for a discrete accelerator, the upper bound of speedups over standard MCMC on the GPU is 39 (image segmentation) and 84 (dense motion estimation) for 320x320 images and 21 and 54 for HD images.

5.5.2.2 Power & Area

We obtain area and power estimates for our proposed RSU from a combination of synthesis of the Verilog circuits using the Synopsys tools, Cacti, and first principles for the RET components [111].

Power. The power for a single RSU-G_1 in 15nm is 3.91mW and is dominated by the electrical power 3.75mW; the RET circuits consume only 0.16mW. A GPU augmented with RSU-G units (3072 in total) consumes 12W of additional power when they are all active. The accelerator with 336 units bounded by 336GB/s DRAM consumes only 1.3W for the RSU-G units. Additional power would be consumed for the memory controller and the control logic. Table 3 lists the power consumption breakdown by RSU-G_1 component.

Table 3: Power consumption for a single RSU-G1.

Power(mW)	**45nm (590MHz)**	**15nm (1GHz)**
Logic	7.20	2.33
RET Circuit	0.16	0.16
LUT	3.92	1.42*
Total	11.28	3.91

*theoretically scaling LUT from 32nm to 15nm [116].

Area. We estimate area of a RSU-G unit by first observing that the SPAD (~1μm^2 [117-119]) and QD-LEDs (~16*25μm^2 [120, 121]) dominate the RET circuit area requirements. The volume of RET network ensemble (~N*20*20*2nm^3) is very small and can reside in a layer above the SPAD. Therefore, we estimate a single RET circuit

requires 400μm^2 and all the RET circuits in an RSU-G_1 unit require 0.0016mm^2. Table 4 lists the area breakdown by RSU-G_1 component.

Table 4: Area for a single RSU-G_1.

Area(μm^2)	**45nm (590MHz)**	**15nm (1GHz)**
Logic	2275	642
RET Circuit	1600	1600
LUT	1798	656*
Total	5673	2898

*theoretically scaling LUT from 32nm to 15nm [116, 122].

5.6 Summary

Despite the theoretical advances of probabilistic algorithms, they can be inefficient on the deterministic hardware that traditional computers use. To develop new hardware that directly supports probabilistic algorithms, we introduce the concept of a RET-based Sampling Unit (RSU), a hybrid CMOS/RET functional unit that efficiently generates samples from parameterized distributions. Specifically, we present the details of a RSU-G unit designed for the MCMC solver for a class of Bayesian Inference problems and the substantial speedups and power savings it brings in different architectures.

Unlike fluorescent taggants, this application of RET networks mainly takes advantage of their efficient random number generation. Nonetheless, these two applications only represent two instances in the vast space of possible applications RET networks can benefit, which will be further discussed in the next chapter.

6. Other Potential Applications

The two applications discussed in this dissertation are based on using RET networks as an entropy source to generate random samples from probability distributions, and demonstrate its great potential in distinct application domains. Specifically, they represent applications that benefit from this molecular-scale entropy source by respectively leveraging its two unique aspects: 1) flexible programmability of probability distributions during fabrication and 2) efficient generation of random numbers through single photon detection. In the application of stochastic computing, this dissertation only focuses on a specific class of inference problems, and the extension to more general MCMC inference engines and other types of probabilistic algorithms (e.g., stochastic neural networks) are future work. Beyond fluorescent taggants and stochastic computing, the application space of RET networks as a programmable entropy source also remains to be explored and may include cryptography.

Meanwhile, there exist alternative ways of using RET networks other than treating them as an entropy source. For example, RET networks may be used to simplify and accelerate probabilistic model checking and rare event simulation by directly exploiting the molecular-scale stochastic process. Probabilistic model checking is an important theoretical framework for evaluating the performance and reliability of a practical system with stochastic behavior, and CTMC is a commonly used stochastic model for mathematically describing such a system by defining its states and associating

its state transitions with probabilities [123, 124]. Based on the CTMC model of a stochastic system, the goal of probabilistic model checking often becomes to analyze the path-based and state-based metrics of interest which can be strictly specified by the temporal logic CSL (Continuous Stochastic Logic) [125]. The two common approaches to the computation of these metrics are 1) numerical probabilistic model checking based on uniformization and 2) stochastic probabilistic model checking based on Monte Carlo simulation and sampling. While the numerical approach generally provides higher accuracy, the stochastic approach can verify models with a larger state space because it requires considerably less memory and scales better.

With RET networks, we may have a third approach that can potentially simplify and accelerate the probabilistic model checking for CTMCs. This approach is based on fabricating a RET network so that the CTMC of its exciton dynamics is directly mapped to the CTMC model of a practical system. With the fabricated RET network, the computation of performance and reliability metrics of the practical system can be carried out by counting the time-resolved fluorescence photons. Specifically, the probability of the system reaching a state by a specific time can be evaluated as the percentage of fluorescence photons detected within a time threshold. Because this approach is based on the physical fabrication and evaluation of a CTMC system, the evaluation process is independent of the size and complexity of the CTMC, and this unique advantage is especially valuable when the events of interest are rare in a stochastic system. To

estimate the probability of rare events, importance sampling is often required as a variance reduction technique and works by replacing the underlying distribution of the system with a biased distribution for generating samples [126, 127]. However, the implementation of this method is highly problem specific and relies on analyzing the mechanism of the system reaching the rare events, which becomes challenging as the size and complexity of the system increase. When a RET network is fabricated and excited by a light source, the excited individual structures in the ensemble simultaneously simulate the same CTMC as different realizations, and the parallelism can potentially reach the scale of Avogadro's number (10^{23}). When the rare event probability is within this range, its value can be conveniently extracted by computing the ratio between the detected photon count and the estimated number of excited structures. This approach to probabilistic model checking is to some extent similar to the concept of 'at-fabrication computation' proposed in previous work [128]. As discussed in Chapter 3, the key challenge of using RET networks for this application is the design of chromophore networks to achieve arbitrary RET networks, which may require innovative and flexible ways of engineering synthetic chromophores.

7. Conclusion

This dissertation introduces the exciton dynamics in a RET network of chromophores as the first molecular-scale process that can be conveniently and accurately programmed to physically implement CTMCs, an important class of stochastic processes. Based on the direct mapping between a RET network and the transition matrix of its CTMC, the stochastic process can be precisely programmed at the molecular scale through the physical geometry of the chromophore network such as chromophore types and the distance between each chromophore pair.

As a molecular-scale photonic device, RET networks has a vast application space in photonics and optoelectronics. This dissertation focuses on using it as a programmable entropy source to generate random samples. Because the fluorescence photons emitted from a RET network follow a phase-type distribution that is configured by its CTMC and phase-type distributions can approximate general distributions, RET networks can be programmed to directly generate true random numbers from different distributions. Used as temporally-coded fluorescent taggants, RET networks can significantly increase the coding capacity of taggant design and improve the reliability of taggant identification even under low light conditions. In the application of stochastic computing, RET-based Sampling Units (RSUs) can be built to efficiently generate samples from parameterized distributions and accelerate probabilistic algorithms.

Appendix A

Below are the matlab scripts for creating the CTMC model of a RET network and performing its steady-state analysis and transient analysis to simulate the time-resolved fluorescence intensity.

```
%%ctmc.m%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%Continuous-Time Markov Chain Model for RET Networks%%
clear;clc;
close all;

global Q

dye_name=[
   'AF405'
   'AF430'
   'AF488'
   'AF546'
   'AF555'
   'AF594'
   'AF610'
   'AF647'
   'AF660'
   'AF680'
   'AF700'
   'AF750'
   'AF790'
   ];

%Förster radius
R0_array=[
   4.892 6.413 6.195 4.465 4.767 3.821 3.966 2.722 4.374 2.718 4.103 3.824
4.411             %AF405
   1.079 4.157 6.272 8.396 8.332 8.803 8.383 7.858 8.970 8.167 8.000 7.038
6.337             %AF430
   0.000 5.830 9.378 11.420 11.569 10.902 9.918 8.390 10.402 8.793 8.202 6.886
6.623           %AF488
   0.000 4.470 5.650 10.807 9.917 13.169 12.570 11.736 13.215 11.911 11.618 10.229
9.108         %AF546
```

```
   0.000 3.457 4.304 8.761 8.274 9.853 9.583 8.908 10.072 9.165 8.953 8.006
7.134              %AF555
   0.000 2.904 3.797 4.947 4.286 11.231 12.538 13.124 14.236 13.594 13.549 12.258
10.889          %AF594
   0.000 2.039 2.659 3.194 2.827 9.558 11.430 12.648 13.372 12.617 12.571 11.177
9.891           %AF610
   0.000 1.343 1.756 2.532 2.140 5.263 6.819 12.733 14.714 14.961 14.886 14.333
13.137          %AF647
   0.000 0.000 0.000 0.000 0.000 3.437 4.753 9.661 12.569 13.512 13.940 13.202
12.132          %AF660
   0.000 0.000 0.000 0.000 0.000 0.000 2.820 8.262 12.010 13.474 14.512 14.601
13.760          %AF680
   0.000 0.000 0.000 0.000 0.000 1.503 3.045 6.746 10.194 11.764 13.204 14.253
13.494          %AF700
   0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 4.631 5.288 7.393 13.428
14.546             %AF750
   0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 1.700 1.827 5.042 10.774
13.125             %AF790
   ];

%Intrinsic Fluorescence Lifetime
tau_array=1e-9*[
   1.2 %AF405
   1.2 %AF430
   4.1 %AF488
   4.1 %AF546
   0.3 %AF555
   3.9 %AF594
   1.2 %AF610
   1.0 %AF647
   1.2 %AF660
   1.2 %AF680
   1.0 %AF700
   0.7 %AF750
   0.7 %AF790
   ];

%Quantum Yield
QY_array=[
   0.4  %AF405
```

```
    0.4  %AF430
    0.92 %AF488
    0.79 %AF546
    0.1  %AF555
    0.66 %AF594
    0.4  %AF610
    0.33 %AF647
    0.37 %AF660
    0.36 %AF680
    0.25 %AF700
    0.12 %AF750
    0.1  %AF790
    ];

kf_array=QY_array./tau_array; %fluorescence rate array
kq_array=1./tau_array-kf_array; %quenching rate array

%peaktime_array=zeros(13,13);
%FWHM_array=zeros(13,13);

% Types of
Chromophores %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
types=[
    2
    6
    6
    6
    6
    6
    6
    12
    ];

% Coordinates of Chromophores %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Separation between adjacent chromophores
sep=[
    0 0
    8.803 0
    11.231 0
    11.231 0
```

```
   11.231 0
   11.231 0
   11.231 0
   12.258 0
   ];

% Coordinates of each chromophore
locs=cumsum(sep);

% Directly assign chromophore coordinates
% locs=1*[
%    0 0
%    10 0
%    20 0
%    30 0
%    40 0
%    50 0
%    60 0
%    70 0
%    ];

% Number of chromophores
num=length(types);

pi_T0 = [1 zeros(1,num-1)];

% Tau's (Intrinsic Fluorescence Lifetimes) %%%%%%%%%%%%%%%%%%%%%%%%
taus = tau_array(types);

% Ro's (Förster Radii) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[id,ia]=meshgrid(types,types);
id=id';ia=ia';
R0=zeros(num,num);
for i=1:num
   for j=1:num
      R0(i,j)=R0_array(id(i,j),ia(i,j));
   end;
end

for i=1:num %no transfer from chromophore to itself
```

```
   R0(i,i)=0;
end

% R matrix (distance between each pair) %%%%%%%%%%%%%%%%%%%%%%%%%%%
R = zeros(num);

for i=1:num
   for j=1:num
         R(i,j)=sqrt((locs(i,1)-locs(j,1))^2+(locs(i,2)-locs(j,2))^2);
   end
end

%Q Matrix%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Qtt = zeros(num,num);              %initiate transfer rate between transient states
Qta = zeros(num,2*num);             %initiate transfer rate from a transient state to an
absorbing state

for i=1:num
   for j=1:num
      if i~=j
         Qtt(i,j)=1/taus(i)*(R0(i,j)/R(i,j))^6; %RET transfer rate for transfer rate between
transient states
      end
   end
end

for i=1:num
   Qta(i,i)=kf_array(types(i));    %Fluorescence
   Qta(i,num+i)=kq_array(types(i));    %Quenching
end

Q=[Qtt Qta
  zeros(2*num,num) zeros(2*num,2*num)
  ];

for i=1:size(Q,1)
   Q(i,i)=-1*sum(Q(i,:));             %Assign values for Qii's
end

%Steady State Analysis
```

```
Qtt=Q(1:num,1:num); %New Qtt Matrix after Qii's assigned
 tau_T=-pi_T0/Qtt;
pi_A=tau_T*Qta;          %Absorption Probability of each absorbing state
theta_T=-tau_T/Qtt;

%Transient Analysis
syms s t;
pi_0 = [pi_T0 zeros(1,2*num)];

options=odeset('RelTol',1e-10,'AbsTol',ones(1,3*num)*1e-10,'InitialStep',1e-12); %ODE
solver
[T,PI] = ode15s(@transfer,linspace(0,50e-9,2^12+1),pi_0,options); %use ODE solver to
solve for the time-resolved state probability of each state [0 25e-9]

cdf=PI(:,2*num);

%plot figures
figure;
hold on;
plot(T,PI(:,2*num),'g-','LineWidth',3);
xlabel('Time (s)','FontSize', 30);
ylabel('Probability','FontSize', 30);
title('Absorption State Probabilities','FontSize', 30);
set(gca,'FontSize',30);
set(gcf,'paperpositionmode','auto');
box on;
hold off;

%Fluorescence Intensity
fdensity=zeros(size(PI)); %differentiate state probabilities to get fluorescence density
(influx of probability into those absorbing states)
for i=2:size(T)-1
   fdensity(i,:)=(PI(i,:)-PI(i-1,:))/(T(i)-T(i-1));
end

%plot figures
figure;
hold on;
plot(T,fdensity(:,2*num)/trapz(T,fdensity(:,2*num)),'g-','LineWidth',3);
xlabel('Time (s)','FontSize', 30);
```

```
ylabel('(Normalized) PDF','FontSize', 30);
title('PDF of The Time-Resolved Fluorescence From Each Chromophore','FontSize', 30);
set(gca,'FontSize',30);
set(gcf,'paperpositionmode','auto');
box on;
hold off;

%%transfer.m%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%Continuout-Time Markov Chain Model for RET Networks%%

function dpi=transfer(t,pi)
global Q
global num_M
global num_S
global num_D

dpi=Q'*pi;

end
```

References

1. R. Brown, "XXVII. A brief account of microscopical observations made in the months of June, July and August 1827, on the particles contained in the pollen of plants; and on the general existence of active molecules in organic and inorganic bodies," Philos. Mag. S. 2 **4**, 161-173 (1828).

2. A. Einstein, "Über die von der molekularkinetischen Theorie der Wärme geforderte Bewegung von in ruhenden Flüssigkeiten suspendierten Teilchen," Ann. Phys. **322**, 549-560 (1905).

3. N. Wiener, "Differential-Space," J. Math. Phys. **2**, 131-174 (1923).

4. F. Horn, "Necessary and sufficient conditions for complex balancing in chemical kinetics," Arch. Ration. Mech. Anal. **49**, 172-186 (1972).

5. M. Feinberg, "Complex balancing in general kinetic systems," Arch. Ration. Mech. Anal. **49**, 187-194 (1972).

6. F. Horn and R. Jackson, "General mass action kinetics," Arch. Ration. Mech. Anal. **47**, 81-116 (1972).

7. D. A. McQuarrie, "Stochastic approach to chemical kinetics," J. Appl. Probab. **4**, 413-478 (1967).

8. N. G. Van Kampen, *Stochastic Processes in Physics and Chemistry* (Elsevier, 1992), Vol. 1.

9. P. Érdi and J. Tóth, *Mathematical Models of Chemical Reactions: Theory and Applications of Deterministic and Stochastic Models* (Manchester University Press, 1989).

10. D. T. Gillespie, "A rigorous derivation of the chemical master equation," Physica A **188**, 404-425 (1992).

11. B. Munsky and M. Khammash, "The finite state projection algorithm for the solution of the chemical master equation," J. Chem. Phys. **124**, 044104 (2006).

12. M. Mateescu, V. Wolf, F. Didier, and T. A. Henzinger, "Fast adaptive uniformisation of the chemical master equation," IET Syst. Biol. **4**, 441-452 (2010).

13. V. Kazeev, M. Khammash, M. Nip, and C. Schwab, "Direct solution of the chemical master equation using quantized tensor trains," PLoS Comput. Biol. **10**, e1003359 (2014).

14. A. Ribeiro, R. Zhu, and S. A. Kauffman, "A general modeling strategy for gene regulatory networks with stochastic dynamics," J. Comput. Biol. **13**, 1630-1639 (2006).

15. D. A. Bratsun, D. N. Volfson, J. Hasty, and L. S. Tsimring, "Non-Markovian processes in gene regulation (Keynote Address)," Proc. SPIE **5845**, 210-219 (2005).

16. G. Karlebach and R. Shamir, "Modelling and analysis of gene regulatory networks," Nat. Rev. Mol. Cell Biol. **9**, 770-780 (2008).

17. B. Valeur and M. N. Berberan-Santos, *Molecular Fluorescence: Principles and Applications, 2nd Edition* (John Wiley & Sons, 2012).

18. S. Wang, A. R. Lebeck, and C. Dwyer, "Nanoscale resonance energy transfer-based devices for probabilistic computing," IEEE Micro **35**, 72-84 (2015).

19. J. R. Lakowicz, *Principles of Fluorescence Spectroscopy* (Springer, 2006), Vol. 1.

20. A. Waggoner, "Fluorescent labels for proteomics and genomics," Curr. Opin. Chem. Biol. **10**, 62-66 (2006).

21. J. Zhang, R. E. Campbell, A. Y. Ting, and R. Y. Tsien, "Creating new fluorescent probes for cell biology," Nat. Rev. Mol. Cell Biol. **3**, 906-918 (2002).

22. W. C. W. Chan and S. M. Nie, "Quantum dot bioconjugates for ultrasensitive nonisotopic detection," Science **281**, 2016-2018 (1998).

23. I. L. Medintz, H. T. Uyeda, E. R. Goldman, and H. Mattoussi, "Quantum dot bioconjugates for imaging, labelling and sensing," Nat. Mater. **4**, 435-446 (2005).

24. A. P. Alivisatos, "Semiconductor clusters, nanocrystals, and quantum dots," Science **271**, 933-937 (1996).

25. I. Hemmila and V. Laitala, "Progress in lanthanides as luminescent probes," J. Fluoresc. **15**, 529-542 (2005).

26. G. D. Hager, M. D. Hill, and K. Yelick, *Opportunities and Challenges for Next Generation Computing (white paper)* (Computing Community Consortium, 2015).

27. K. S. Trivedi, *Probability and Statistics with Reliability, Queueing, and Computer Science Applications* (John Wiley & Sons, 2008).

28. E. Wetterskog, M. Agthe, A. Mayence, J. Grins, D. Wang, S. Rana, A. Ahniyaz, G. Salazar-Alvarez, and L. Bergström, "Precise control over shape and size of iron oxide nanocrystals suitable for assembly into ordered particle arrays," Sci. Technol. Adv. Mater. **15**, 055010 (2014).

29. T. A. Pham, F. Song, M.-T. Nguyen, and M. Stohr, "Self-assembly of pyrene derivatives on Au(111): substituent effects on intermolecular interactions," Chem. Commun. **50**, 14089-14092 (2014).

30. P. W. K. Rothemund, "Folding DNA to create nanoscale shapes and patterns," Nature **440**, 297-302 (2006).

31. C. Pistol and C. Dwyer, "Scalable, low-cost, hierarchical assembly of programmable DNA nanostructures," Nanotechnology **18**, 125305-125309 (2007).

32. C. LaBoda, H. Duschl, and C. L. Dwyer, "DNA-enabled integrated molecular systems for computation and sensing," Acc. Chem. Res. **47**, 1816-1824 (2014).

33. C. Pistol, V. Mao, V. Thusu, A. R. Lebeck, and C. Dwyer, "Encoded multichromophore response for simultaneous label-free detection," Small **6**, 843-850 (2010).

34. C. Pistol, C. Dwyer, and A. R. Lebeck, "Nanoscale optical computing using resonance energy transfer logic," IEEE Micro **28**, 7-18 (2008).

35. U. Resch-Genger, M. Grabolle, S. Cavaliere-Jaricot, R. Nitschke, and T. Nann, "Quantum dots versus organic dyes as fluorescent labels," Nat. Methods **5**, 763-775 (2008).

36. N. C. Shaner, P. A. Steinbach, and R. Y. Tsien, "A guide to choosing fluorescent proteins," Nat. Methods **2**, 905-909 (2005).

37. L. Adleman, "Molecular computation of solutions to combinatorial problems," Science **266**, 1021-1024 (1994).

38. N. C. Seeman, "Nucleic acid junctions and lattices," J. Theor. Biol. **99**, 237-247 (1982).

39. J. Chen and N. C. Seeman, "Synthesis from DNA of a molecule with the connectivity of a cube," Nature **350**, 631-633 (1991).

40. D. Han, S. Pal, J. Nangreave, Z. Deng, Y. Liu, and H. Yan, "DNA origami with complex curvatures in three-dimensional space," Science **332**, 342-346 (2011).

41. Veneziano, Rémi, Ratanalert, Sakul, Zhang, Kaiming, Zhang, Fei, Yan, Hao, Chiu, Wah, and M. Bathe, "Designer nanoscale DNA assemblies programmed from the top down," Science (2016).

42. L. Qian and E. Winfree, "Scaling up digital circuit computation with DNA strand displacement cascades," Science **332**, 1196-1201 (2011).

43. L. Qian, E. Winfree, and J. Bruck, "Neural network computation with DNA strand displacement cascades," Nature **475**, 368-372 (2011).

44. D. Y. Zhang and G. Seelig, "Dynamic DNA nanotechnology using strand-displacement reactions," Nat Chem **3**, 103-113 (2011).

45. D. Soloveichik, G. Seelig, and E. Winfree, "DNA as a universal substrate for chemical kinetics," Proc. Natl. Acad. Sci. U.S.A. **107**, 5393-5398 (2010).

46. D. Soloveichik, M. Cook, E. Winfree, and J. Bruck, "Computation with finite stochastic chemical reaction networks," Nat. Computing **7**, 615-633 (2008).

47. M. Cook, D. Soloveichik, E. Winfree, and J. Bruck, "Programmability of chemical reaction networks," in *Algorithmic Bioprocesses*, A. Condon, D. Harel, N. J. Kok, A. Salomaa, and E. Winfree, eds. (Springer Berlin Heidelberg, Berlin, Heidelberg, 2009), pp. 543-584.

48. M. Stipčević and Ç. K. Koç, "True Random Number Generators," in *Open Problems in Mathematics and Computational Science* (Springer, 2014), pp. 275-315.

49. M. Wahl, M. Leifgen, M. Berlin, T. Röhlicke, H.-J. Rahn, and O. Benson, "An ultrafast quantum random number generator with provably bounded output bias based on photon arrival time measurements," Appl. Phys. Lett. **98**, 171105 (2011).

50. G. Juzeliūnas and D. L. Andrews, "Quantum Electrodynamics of Resonance Energy Transfer," in *Advances in Chemical Physics* (John Wiley & Sons, Inc., 2007), pp. 357-410.

51. A. Salam and A. Salam, "Resonant Transfer Of Energy," in *Molecular Quantum Electrodynamics* (John Wiley & Sons, Inc., 2009), pp. 139-174.

52. D. R. Cox, "A use of complex probabilities in the theory of stochastic processes," Math. Proc. Cambridge Philos. Soc. **51**, 313-319 (1955).

53. M. F. Neuts, *Matrix-geometric Solutions in Stochastic Models: An Algorithmic Approach* (Dover Publications, 1981).

54. T. Osogami and M. Harchol-Balter, "A closed-form solution for mapping general distributions to minimal PH distributions," in *Computer Performance Evaluation. Modelling Techniques and Tools: 13th International Conference*, (Springer Berlin Heidelberg, 2003), 200-217.

55. M. Olsson, "The EMpht-programme," (Chalmers University of Technology, and Göteborg University, Sweden, 1998).

56. W. Becker, *The bh TCSPC Handbook, 6th edition* (Becker & Hickl, 2015).

57. A. Bobbio, A. Horváth, and M. Telek, "Matching three moments with minimal acyclic phase type distributions," Stoch. Model **21**, 303-326 (2005).

58. M. A. Johnson and M. R. Taaffe, "Matching moments to phase distributions: Mixtures of erlang distributions of common order," Comm. Statist. Stochastic Models **5**, 711-743 (1989).

59. S. Asmussen, xf, ren, O. Nerman, and O. Marita, "Fitting phase-type distributions via the EM algorithm," Scand. J. Stat. **23**, 419-441 (1996).

60. A. Thummler, P. Buchholz, and M. Telek, "A novel approach for phase-type fitting with the EM algorithm," IEEE Trans. Dependable Secure Comput. **3**, 245-258 (2006).

61. H. Okamura, T. Dohi, and K. S. Trivedi, "A refined EM algorithm for PH distributions," Perform. Eval. **68**, 938-954 (2011).

62. L. u. Korenčiak, J. Krčál, and V. Řehák, "Dealing with Zero Density Using Piecewise Phase-Type Approximation," in *Computer Performance Engineering: 11th European Workshop, EPEW 2014, Florence, Italy, September 11-12, 2014. Proceedings*, A. Horváth and K. Wolter, eds. (Springer International Publishing, Cham, 2014), pp. 119-134.

63. V. K. Mansinghka and E. Jonas, "Building fast Bayesian computing machines out of intentionally stochastic, digital parts," Computing Research Repository (ArXiv) **abs/1402.4914**(2014).

64. S. Wang, R. Vyas, and C. Dwyer, "Fluorescent taggants with temporally coded signatures," Opt. Express **24**, 15528-15545 (2016).

65. M. Sauer, J. Arden-Jacob, K. H. Drexhage, F. Gobel, U. Lieberwirth, K. Muhlegger, R. Muller, J. Wolfrum, and C. Zander, "Time-resolved identification of individual mononucleotide molecules in aqueous solution with pulsed semiconductor lasers," Bioimaging **6**, 14-24 (1998).

66. Y. Lu, J. Lu, J. Zhao, J. Cusido, F. M. Raymo, J. Yuan, S. Yang, R. C. Leif, Y. Huo, J. A. Piper, J. P. Robinson, E. M. Goldys, and D. Jin, "On-the-fly decoding luminescence lifetimes in the microsecond region for lanthanide-encoded suspension arrays," Nat. Commun. **5**, 3741 (2014).

67. U. Lieberwirth, J. Arden-Jacob, K. H. Drexhage, D. P. Herten, R. Muller, M. Neumann, A. Schulz, S. Siebert, G. Sagner, S. Klingel, M. Sauer, and J. Wolfrum, "Multiplex dye DNA sequencing in capillary gel electrophoresis by diode laser-based time-resolved fluorescence detection," Anal. Chem. **70**, 4771-4779 (1998).

68. W. F. Hug, R. Bhartia, and A. Tsapin, "Water & surface contamination monitoring using deep UV laser induced native fluorescence and Raman spectroscopy," Proc. SPIE **6378**, 63780S (2006).

69. G. P. Asner and R. E. Martin, "Airborne spectranomics: mapping canopy chemical and taxonomic diversity in tropical forests," Front. Ecol. Environ. **7**, 269-276 (2009).

70. G. P. Asner, D. E. Knapp, T. Kennedy-Bowdoin, M. O. Jones, R. E. Martin, J. Boardman, and R. F. Hughes, "Invasive species detection in Hawaiian rainforests using airborne imaging spectroscopy and LiDAR," Remote Sens. Environ. **112**, 1942-1955 (2008).

71. A. H. Omar, D. M. Winker, C. Kittaka, M. A. Vaughan, Z. Liu, Y. Hu, C. R. Trepte, R. R. Rogers, R. A. Ferrare, K.-P. Lee, R. E. Kuehn, and C. A. Hostetler, "The CALIPSO automated aerosol classification and lidar ratio selection algorithm," J. Atmos. Oceanic Technol. **26**, 1994-2014 (2009).

72. G. M. Williams, Jr., T. Allen, C. Dupuy, T. Novet, and D. Schut, "Optically coded nanocrystal taggants and optical frequency IDs," Proc. SPIE **7673**, 76730M (2010).

73. G. McLachlan and T. Krishnan, *The EM Algorithm and Extensions* (John Wiley & Sons, 2007), Vol. 382.

74. "ID100 single photon detection module with high timing resolution and low dark count rate" (ID Quantique), retrieved http://www.idquantique.com/wordpress/wp-content/uploads/id100-specs.pdf.

75. Y. Wang, B. R. Rae, R. K. Henderson, Z. Gong, J. Mckendry, E. Gu, M. D. Dawson, G. A. Turnbull, and I. D. W. Samuel, "Ultra-portable explosives sensor based on a CMOS fluorescence lifetime analysis micro-system," AIP Advances **1**, 032115 (2011).

76. D. Tyndall, "CMOS system for high throughput fluorescence lifetime sensing using time correlated single photon counting," Ph.D. thesis (The University of Edinburgh, 2013).

77. Ž. Bajzer, T. M. Therneau, J. C. Sharp, and F. G. Prendergast, "Maximum likelihood method for the analysis of time-resolved fluorescence decay curves," Eur. Biophys. J. **20**, 247-262 (1991).

78. M. Köllner and J. Wolfrum, "How many photons are necessary for fluorescence-lifetime measurements?," Chem. Phys. Lett. **200**, 199-204 (1992).

79. M. Köllner, "How to find the sensitivity limit for DNA sequencing based on laser-induced fluorescence," Appl. Opt. **32**, 806-820 (1993).

80. M. Kollner, A. Fischer, J. ArdenJacob, K. H. Drexhage, R. Muller, S. Seeger, and J. Wolfrum, "Fluorescence pattern recognition for ultrasensitive molecule identification: Comparison of experimental data and theoretical approximations," Chem. Phys. Lett. **250**, 355-360 (1996).

81. I. M. Warner, G. D. Christian, E. R. Davidson, and J. B. Callis, "Analysis of multicomponent fluorescence data," Anal. Chem. **49**, 564-573 (1977).

82. N. Keshava and J. F. Mustard, "Spectral unmixing," IEEE Signal Process Mag. **19**, 44-57 (2002).

83. D. C. Heinz and C. I. Chang, "Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery," IEEE Trans. Geosci. Remote Sens. **39**, 529-545 (2001).

84. D. Lognoli, G. Cecchi, I. Mochi, L. Pantani, V. Raimondi, R. Chiari, T. Johansson, P. Weibring, H. Edner, and S. Svanberg, "Fluorescence lidar imaging of the cathedral and baptistery of Parma," Appl. Phys. B: Lasers Opt. **76**, 457-465 (2003).

85. L. Palombi, D. Alderighi, G. Cecchi, V. Raimondi, G. Toci, and D. Lognoli, "A fluorescence LIDAR sensor for hyper-spectral time-resolved remote sensing and mapping," Opt. Express **21**, 14736-14746 (2013).

86. V. Raimondi, D. Lognoli, and L. Palombi, "A fluorescence lidar combining spectral, lifetime and imaging capabilities for the remote sensing of cultural heritage assets," Proc. SPIE **9245**, 92450K (2014).

87. L. Palombi, D. Lognoli, and V. Raimondi, "Fluorescence LIDAR remote sensing of oils: merging spectral and time-decay measurements," Proc. SPIE **8887**, 88870F (2013).

88. V. K. Mansinghka, "Natively Probabilistic Computation," (MIT, 2009).

89. T. Minka, J. Winn, J. Guiver, S. Webster, Y. Zaykov, B. Yangel, A. Spengler, and J. Bronskill, Infer.NET 2.6, Microsoft Research Cambridge, 2014.

90. R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother, "A comparative study of energy minimization methods for markov random fields with smoothness-based priors," IEEE Trans. Pattern Anal. Mach. Intell. **30**, 1068-1080 (2008).

91. M. Mitzenmacher and E. Upfal, *Probability and Computing: Randomized Algorithms and Probabilistic Analysis* (Cambridge University Press, 2005).

92. K. P. Murphy, *Machine Learning: a Probabilistic Perspective* (MIT press, 2012).

93. L. Devroye, *Non-uniform Random Variate Generation* (Springer-Verlag, 1986).

94. "C++ Pseudo-random number generation library", retrieved http://en.cppreference.com/w/cpp/numeric/random.

95. L. Luan, R. D. Evans, N. M. Jokerst, and R. B. Fair, "Integrated optical sensor in a digital microfluidic platform," IEEE Sens. J. **8**, 628-635 (2008).

96. L. Luan, M. W. Royal, R. Evans, R. B. Fair, and N. M. Jokerst, "Chip scale optical microresonator sensors integrated with embedded thin film photodetectors on electrowetting digital microfluidics platforms," IEEE Sens. J. **12**, 1794-1800 (2012).

97. T. P. Minka, "Expectation propagation for approximate Bayesian inference," in *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, (Morgan Kaufmann Publishers Inc., 2001), 362-369.

98. J. M. Winn and C. M. Bishop, "Variational message passing," J. Mach. Learn. Res., 661-694 (2005).

99. "Working with different inference algorithms", retrieved http://research.microsoft.com/en-us/um/cambridge/projects/infernet/docs/Working%20with%20different%20inference%20algorithms.aspx.

100. M. Aono, M. Naruse, S.-J. Kim, M. Wakabayashi, H. Hori, M. Ohtsu, and M. Hara, "Amoeba-Inspired Nanoarchitectonic Computing: Solving Intractable Computational Problems Using Nanoscale Photoexcitation Transfer Dynamics," Langmuir **29**, 7557-7564 (2013).

101. M. Naruse, M. Aono, and S.-J. Kim, "Nanoscale Photonic Network for Solution Searching and Decision Making Problems," IEICE Trans. Commun. **E96.B**, 2724-2732 (2013).

102. "Intel® Digital Random Number Generator (DRNG) Software Implementation Guide", retrieved https://software.intel.com/sites/default/files/managed/4d/91/DRNG_Software_Implementation_Guide_2.0.pdf.

103. "Verilog Implementation of AES As Specified in NIST FIPS 197", retrieved https://github.com/secworks/aes.

104. L. N. Chakrapani, B. E. S. Akgul, S. Cheemalavagu, P. Korkmaz, K. V. Palem, and B. Seshasayee, "Ultra-Efficient (Embedded) SOC Architectures based on Probabilistic CMOS (PCMOS) Technology," in *Design, Automation and Test in Europe, 2006. DATE '06. Proceedings*, 2006), 1-6.

105. K. V. Palem, "Energy aware computing through probabilistic switching: a study of limits," IEEE Trans. Comput. **54**, 1123-1137 (2005).

106. R. S. Amant, A. Yazdanbakhsh, J. Park, B. Thwaites, H. Esmaeilzadeh, A. Hassibi, L. Ceze, and D. Burger, "General-Purpose Code Acceleration with Limited-Precision Analog Computation," in *Proceedings of the 41st Annual International Symposium on Computer Architecture (ISCA)*, (IEEE, 2014), 505-516.

107. H. Esmaeilzadeh, A. Sampson, L. Ceze, and D. Burger, "Neural Acceleration for General-Purpose Approximate Programs," in *Proceedings of the 2012 45th Annual IEEE/ACM International Symposium on Microarchitecture*, (IEEE Computer Society, 2012), 449-460.

108. J. Bornholt, T. Mytkowicz, and K. S. McKinley, "Uncertain<T>: a first-order type for uncertain data," in *Proceedings of 19th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, (ACM, 2014), 51-66.

109. S. Z. Li and S. Singh, *Markov Random Field Modeling in Image Analysis* (Springer, 2009), Vol. 26.

110. S. Geman and C. Graffigne, "Markov random field image models and their applications to computer vision," in *Proceedings of the International Congress of Mathematicians*, (AMS, Providence, RI, 1986), 2.

111. S. Wang, X. Zhang, Y. Li, R. Bashizade, S. Yang, C. Dwyer, and A. R. Lebeck, "Accelerating markov random field inference using molecular optical Gibbs sampling units," in *Proceedings of the 43nd Annual International Symposium on Computer Architecture (ISCA)*, (ACM, 2016), .

112. T. Szirányi, J. Zerubia, L. Czúni, D. Geldreich, and Z. Kato, "Image segmentation using Markov random field model in fully parallel cellular network architectures," Real-Time Imaging **6**, 195-211 (2000).

113. S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," IEEE Trans. Pattern Anal. Mach. Intell., 721-741 (1984).

114. M. F. Tappen and W. T. Freeman, "Comparison of graph cuts with belief propagation for stereo, using identical MRF parameters," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, (IEEE, 2003), 900-906.

115. J. Konrad and E. Dubois, "Bayesian estimation of motion vector fields," IEEE Trans. Pattern Anal. Mach. Intell. **14**, 910-927 (1992).

116. S. Thoziyoor, N. Muralimanohar, J. Ahn, and N. Jouppi, "Cacti 5.3," HP Laboratories, Palo Alto, CA (2008).

117. S. Mandai, M. W. Fishburn, Y. Maruyama, and E. Charbon, "A wide spectral range single-photon avalanche diode fabricated in an advanced 180 nm CMOS technology," Opt. Express **20**, 5849-5857 (2012).

118. D. Palubiak, M. M. El-Desouki, O. Marinov, M. Deen, and Q. Fang, "High-speed, single-photon avalanche-photodiode imager for biomedical applications," IEEE Sens. J. **11**, 2401-2412 (2011).

119. S. Assefa, F. Xia, and Y. A. Vlasov, "Reinventing germanium avalanche photodetector for nanophotonic on-chip optical interconnects," Nature **464**, 80-84 (2010).

120. M. T. Hill and M. C. Gather, "Advances in small lasers," Nat. Photonics **8**, 908-918 (2014).

121. G. Shambat, B. Ellis, J. Petykiewicz, M. A. Mayer, A. Majumdar, T. Sarmiento, J. S. Harris, E. E. Haller, and J. Vuckovic, "Electrically Driven Photonic Crystal Nanocavity Devices," IEEE J. Sel. Top. Quantum Electron. **18**, 1700-1710 (2012).

122. S. Taejoong, R. Woojin, J. Jonghoon, Y. Giyong, P. Jaeho, P. Sunghyun, B. Kang-Hyun, B. Sanghoon, O. Sang-Kyu, J. Jinsuk, K. Sungbong, K. Gyuhong, K. Jintae, L. YoungKeun, K. Kee Sup, S. Sang-Pil, Y. Jong Shik, and C. Kyu-Myung, "13.2 A 14nm FinFET 128Mb 6T SRAM with VMIN-enhancement techniques for low-power applications," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014 IEEE International*, 2014), 232-233.

123. C. Baier, B. Haverkort, H. Hermanns, and J. P. Katoen, "Model-checking algorithms for continuous-time Markov chains," IEEE Trans. Software Eng. **29**, 524-541 (2003).

124. H. L. S. Younes, M. Kwiatkowska, G. Norman, and D. Parker, "Numerical vs. statistical probabilistic model checking," Int. J. Software Tools Technol. Trans. **8**, 216-228 (2006).

125. A. Aziz, K. Sanwal, V. Singhal, and R. Brayton, "Model-checking continuous-time Markov chains," ACM Trans. Comput. Logic **1**, 162-170 (2000).

126. B. Barbot, S. Haddad, and C. Picaronny, "Importance sampling for model checking of continuous time Markov chains," in *Proceedings of the 4th International Conference on Advances in System Simulation (SIMUL'12)*, (XPS, 2012), 30-35.

127. Z. I. Botev, P. L'Ecuyer, and B. Tuffin, "Markov chain importance sampling with applications to rare event probability estimation," Stat. Comput. **23**, 271-285 (2013).

128. C. Dwyer, A. R. Lebeck, and D. J. Sorin, "Self-assembled architectures and the temporal aspects of computing," Computer **38**, 56-64 (2005).

Biography

Siyang Wang was born on October 19th, 1988 in Jiangxi Province, China. He received his Bachelor of Science in Microelectronics from Shanghai Jiao Tong University in 2010 and Master of Science in Electrical and Computer Engineering from Duke University in 2012. His research interests include molecular computing, probabilistic computing, resonance energy transfer and DNA self-assembly. His Ph.D thesis explores the molecular-scale stochastic process based on resonance energy transfer networks and its applications in fluorescent taggants and stochastic computing.

Publications:

[1] S. Wang, A. R. Lebeck, and C. Dwyer, "Nanoscale resonance energy transfer-based devices for probabilistic computing," IEEE Micro **35**, 72-84 (2015).

[2] S. Wang, R. Vyas, and C. Dwyer, "Fluorescent taggants with temporally coded signatures," Opt. Express **24**, 15528-15545 (2016).

[3] S. Wang, X. Zhang, Y. Li, R. Bashizade, S. Yang, C. Dwyer, and A. R. Lebeck, "Accelerating markov random field inference using molecular optical Gibbs sampling units," in *Proceedings of the 43nd Annual International Symposium on Computer Architecture (ISCA)*, (ACM, 2016), .

[4] C. Dwyer, A. Rallapalli, M. Mottaghi, and S. Wang, "DNA Self-Assembled Nanostructures for Resonance Energy Transfer Circuits," in *Nanophotonic Information Physics* (Springer, 2014), pp. 41-65.