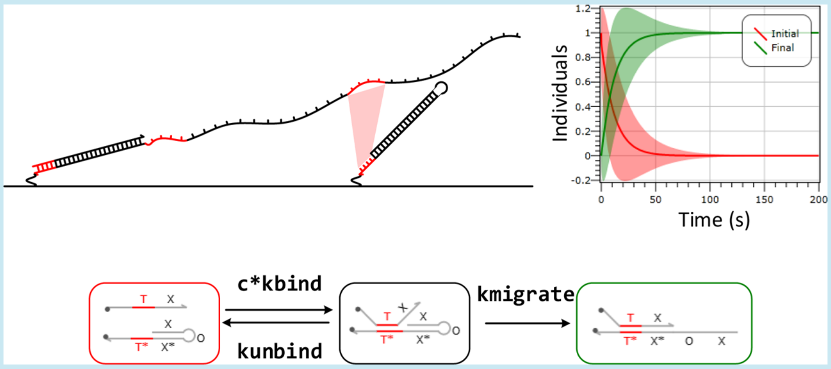
ACS Synthetic Biology
Research Article
pubs.acs.org/synthbio

Probabilistic Analysis of Localized DNA Hybridization Circuits

Neil Dalchau,[†,⊥] Harish Chandran,[‡,⊥,#] Nikhil Gopalkrishnan,[‡,§] Andrew Phillips,[*,†] and John Reif[*,‡]

[†]Microsoft Research, Cambridge CB1 2FB, United Kingdom
[‡]Department of Computer Science, Duke University, Durham, North Carolina 27708, United States
[§]Wyss Institute, Harvard University, Boston, Massachusetts 02115, United States

Supporting Information



ABSTRACT: Molecular devices made of nucleic acids can perform complex information processing tasks at the nanoscale, with potential applications in biofabrication and smart therapeutics. However, limitations in the speed and scalability of such devices in a well-mixed setting can significantly affect their performance. In this article, we propose designs for localized circuits involving DNA molecules that are arranged on addressable substrates and interact via hybridization reactions. We propose designs for localized elementary logic circuits, which we compose to produce more complex devices, including a circuit for computing the square root of a four bit number. We develop an efficient method for probabilistic model checking of localized circuits, which we implement within the Visual DSD design tool. We use this method to prove the correctness of our circuits with respect to their functional specifications and to analyze their performance over a broad range of local rate parameters. Specifically, we analyze the extent to which our localized designs can overcome the limitations of well-mixed circuits, with respect to speed and scalability. To provide an estimate of local rate parameters, we propose a biophysical model of localized hybridization. Finally, we use our analysis to identify constraints in the rate parameters that enable localized circuits to retain their advantages in the presence of unintended interferences between strands.

Molecular devices made of nucleic acids can perform complex information processing tasks at the nanoscale and interface directly with biomolecular components. As a result, such devices show great potential for use in applications ranging from biofabrication to smart therapeutics. Recently, DNA strand displacement[1] has been proposed as a promising approach for implementing molecular scale information processing. It involves the displacement of a single strand of DNA from a double-stranded template by an incoming strand and relies on DNA hybridization, the noncovalent binding of two complementary DNA sequences to form a single duplex structure. Recent theoretical work[2] demonstrated the potential for DNA strand displacement to implement a broad range of complex computations, including any behavior that can be expressed as a chemical reaction network. Over the past decade, steady progress has been made toward the experimental implementation of DNA strand displacement devices that are autonomous, error-resilient, and scalable.[1] Examples include catalyzed formation of branched structures, autocatalytic exponential amplifiers, and bipedal walkers,[3] together with the use of scalable seesaw gates[4] to perform a square root calculation[5] and to implement artificial neural networks.[6] More recently, catalytic strand displacement circuits were used to implement a distributed consensus network.[7] However, when implementing complex strand displacement devices in a well-mixed setting, speed and scalability have been identified as significant challenges.[5] Since interacting strands are freely floating in solution, interaction rates are primarily limited by the time taken for strands to encounter each other via the relatively slow process of diffusion. For example, the well-mixed square root circuit[5] required about 9 h for its slowest output to reach 75% completion, whereas the well-mixed consensus circuit[7] required 15 h to converge. Furthermore, spurious binding was observed to slow the reaction rates of the square root circuit[5] and to decrease the effectiveness of the circuit's

Special Issue: IWBDA 2014

Received: October 22, 2014
Published: July 1, 2015

DOI: 10.1021/acssynbio.5b00044
ACS Synth. Biol. 2015, 4, 898−913

components. Improving sequence design and using lower concentrations were proposed as partial solutions; however, reduced concentrations also result in reduced speed, leading to a trade-off between speed and scalability.

To overcome the limitations of well-mixed devices, a transition from solution-phase circuitry to circuitry organized on a surface such as DNA origami was proposed.[5] Localized designs would enable adjacent DNA gates to interact without diffusion, limiting spurious interactions to immediate neighbors and allowing sequences to be safely reused in spatially separated locations.[5] An extended abstract of the present work[8] described how addressable DNA substrates could be used as building blocks for localized hybridization circuits to overcome many of the limitations of well-mixed designs. A number of localized hybridization schemes have since been implemented experimentally, including a scheme in which hairpin strands are tethered to a surface and the interactions between strands are mediated by diffusible fuel hairpins.[9] This has advantages of circuit scalability, since separate circuits can execute in spatially distinct locations. However, the use of diffusible fuel limits the potential speedup of these circuits. An alternative scheme was used to study the robustness of localized DNA strand displacement cascades.[10] In this scheme,[10] sender and receiver gates were localized to a DNA origami platform, and communication between gates was achieved by means of diffusible intermediate strands. Although the circuits operated by means of diffusion, the close proximity of sender and receiver gates still resulted in increased speedup, which varied depending on the distance between gates. Another set of localized hybridization schemes are molecular walkers and robots.[11−16] Kinetic studies of such systems have revealed that molecular transport can operate efficiently, even when relying on enzymes in addition to toehold-mediated strand displacement for propagation, providing useful observations for the parametrization of alternative localized schemes.[14] The demonstration of using DNA origami as a microarray is also a promising avenue for analyzing localized hybridization rules, although it has currently been applied only to interactions between a diffusible input and a fixed, although measurable, docking site.[17] Finally, we note that naturally occurring biological systems have solved the problems of speed and scalability of computation through spatial organization, for example, by using intracellular compartments to carry out dedicated tasks, using scaffolded enzyme cascades for signal propagation,[18] or using membrane-bound receptors to perform detection on a surface rather than within a volume.

Methods and software have proven to be valuable tools for the design and analysis of DNA strand displacement systems.[1] For instance, a compiler for translating an arbitrary feed forward digital logic circuit to a strand displacement system expressed as seesaw gates has been proposed.[5] In addition, Visual DSD[19,20] has been proposed as a tool for the design and analysis of DNA strand displacement systems and has been used to aid the design of a number of systems implemented experimentally.[5−7,21] However, when considering the specific case of localized DNA circuits, care must be taken to ensure that appropriate modeling approaches are used. In such cases, the role of stochasticity becomes important because molecules interact at low copy numbers. Consequently, techniques for the analysis of Markov processes must be considered for model analysis to be relevant. Probabilistic model checking of a stochastic system involves the analysis of all possible executions of the system to determine the probability of a specified behavior being observed. This is achieved by considering the possible states of a closed system and the rates of transition between each state, which together form a continuous-time Markov chain (CTMC). Logical queries about the system can be checked such as the mean time until a certain state is reached, which might represent the completion of a local computation, or simply the expected copy number of a particular molecule at a specific time. The PRISM software tool[22] is the most established tool for testing logical queries by probabilistic model checking. In recent work, PRISM was used to analyze localized walker circuits[23] and was also used in combination with Visual DSD to perform probabilistic analysis of well-mixed strand displacement circuits.[24] To facilitate the efficient analysis of complex strand displacement systems, we sought to integrate probabilistic model checking within the Visual DSD tool directly. There were two major reasons for adopting this strategy, instead of using the previously described workflow involving PRISM.[24] The first was that by having the probabilistic analysis functionality within the same tool the overall workflow was greatly simplified. This enabled a more streamlined approach to analyzing modifications to a given model. The second was that by changing our calculation strategy we were able to optimize the computation of probabilistic queries at multiple time points. To calculate so-called transient probabilities, PRISM uses a uniformization method, which relies on a discrete time conversion of the CTMC to calculate the probability of being in a particular state at a particular time. When requesting multiple time points, the process is applied multiple times, each of which comes with a setup cost. The strategy we adopted relies on numerical integration (see the Supporting Information), which propagates the solution at a previous time point to the next one, removing the computational cost of requesting successive transient probabilities. While uniformization often provides a more efficient and numerically stable routine for single calculations,[25] we find that adaptive step-size numerical integration is orders of magnitude more efficient for probability calculations over time, at least for the systems studied here.

In this article, we present a method for the probabilistic analysis of localized hybridization circuits. To illustrate the generality of the method with respect to specific biophysical parameters, we perform probabilistic analysis of an elementary localized device for a broad range of local concentrations, where different local concentrations are representative of different biophysical implementations. We then propose localized hybridization designs for elementary logic circuits and use probabilistic analysis to characterize the performance of these circuits as a function of local concentration. We further consider spurious leak interactions between strands to determine their potential impact on circuit performance. We identify a trade-off between the speedup of circuit completion and the speedup of leak interactions and use this to determine an optimal value of local concentration. To determine how this concentration could be achieved in a physical implementation, we propose a biophysical model of tethered hybridization. We then use our elementary localized circuits to produce more complex devices, including a circuit that computes the square root of a four bit number. We use our probabilistic analysis method to prove the correctness of our designs with respect to their functional specifications and to analyze their performance over time. We then consider the use of localized circuits in the presence of diffusible inputs, which corresponds to an important experimental context. We use our analysis to identify

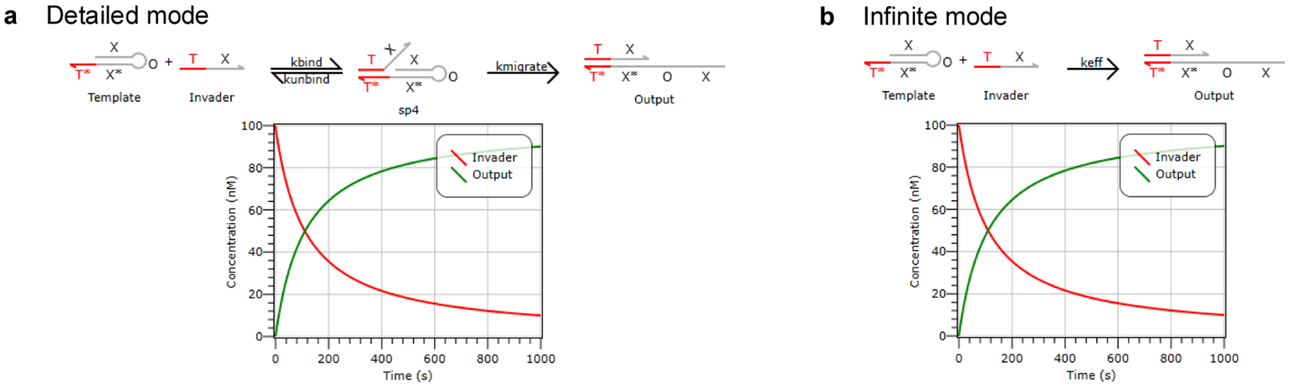


Figure 1. Comparison of compilation modes in Visual DSD. (a) Chemical reaction network (CRN) and ODE simulation for an elementary strand displacement circuit in Detailed mode, in which binding, unbinding, and migration are represented as separate steps. We assume 100 nM of template and invader strands initially, with rates $k_{\text{bind}} = 10^{-3}$ nM^{-1} s^{-1}, $k_{\text{unbind}} = 10$ s^{-1}, and $k_{\text{migrate}} = 1$ s^{-1}. (b) CRN and ODE simulation for the elementary strand displacement circuit in Infinite mode, in which strand displacement is assumed to take place in a single step, with rate $k_{\text{eff}} = k_{\text{bind}} \cdot k_{\text{migrate}}/(k_{\text{migrate}} + k_{\text{unbind}}) = k_{\text{bind}}/11$.

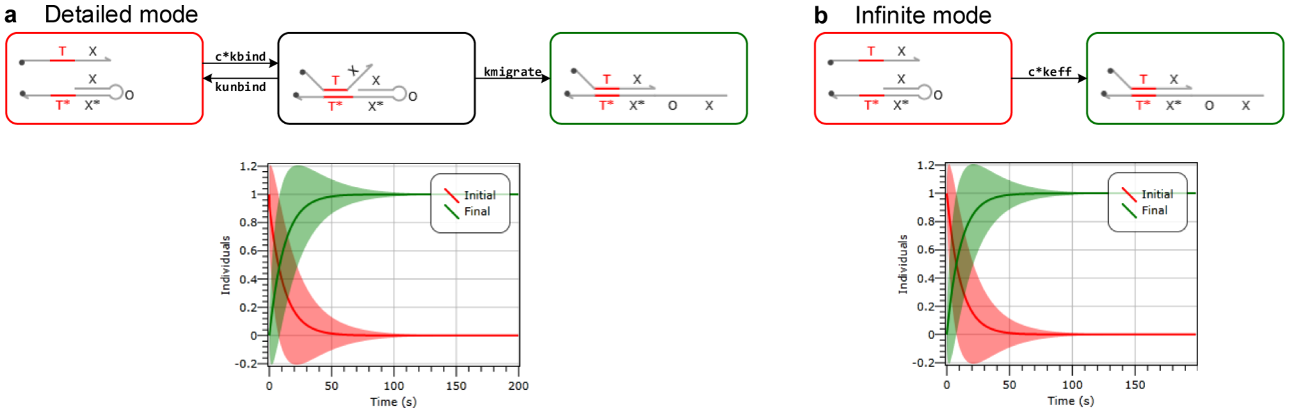


Figure 2. Comparison of compilation modes for localized strand displacement circuits. (a) Continuous-time Markov chain (CTMC) and CME analysis for a localized version of the elementary strand displacement circuit from Figure 1a in Detailed mode. Here, both the invader and template are tethered to the same substrate, in close proximity to one another. The localized binding rate is given by $c \cdot k_{\text{bind}}$, where c is the local concentration, which quantifies the proximity of the two strands. Here, we assume $c = 10^3$ nM. The analysis denotes the probability of the circuit being in a particular state at a given time (solid lines), which corresponds to the expected proportion of circuits in that state, together with the standard deviation for each state (shaded region). (b) CTMC and CME analysis for a localized version of the elementary strand displacement circuit from Figure 1b in Infinite mode.

limitations of circuit performance in this context and propose a number of design improvements to overcome these limitations. Finally, we propose a method for organizing DNA circuits on addressable DNA substrates by parallel synthesis that is experimentally feasible and scales to a large number of gates.

RESULTS

A Method for Probabilistic Analysis of Localized Circuits. In this section, we present our method for the probabilistic analysis of localized hybridization circuits. To introduce the method, we compare the deterministic simulation of an elementary strand displacement device in a well-mixed setting with the probabilistic analysis of the same device in a localized setting. To illustrate the generality of the method with respect to specific biophysical implementations, we perform probabilistic analysis of the localized circuit for a broad range of local concentrations.

DNA strand displacement[1] is characterized by an invading single strand of DNA displacing an incumbent strand bound to a template. The template has a toehold, or short single-stranded region, which is complementary to a region on the invading strand. The invading strand binds reversibly to the template and eventually displaces the incumbent strand via a kinetic process of branch migration, modeled as a one-dimensional random walk.[26,27] An elementary strand displacement device is shown in Figure 1a, in which the template takes the form of a partially double-stranded molecule with a hairpin loop. The invading strand ⟨TX⟩ binds to the template strand ⟨XOX^*T^*⟩ on the toehold T, where T^* denotes the sequence complementary to T. The two strands bind with rate k_{bind} and unbind with rate k_{unbind}. The branch migration occurs with rate k_{migrate} and eventually opens the hairpin to reveal the output O. In a well-mixed setting, the Visual DSD tool is used to automatically generate a chemical reaction network (CRN) of this device. Here, we consider CRNs generated using two alternative modes, a Detailed mode CRN (Figure 1a), which represents toehold binding, unbinding, and branch migration as separate steps, and an Infinite mode CRN (Figure 1b), which represents strand displacement as a single, merged step. These modes correspond, respectively, to previously proposed three- and one-step models.[26] Following the approach of Zhang and Winfree,[26] we compute the effective rate of the merged step as

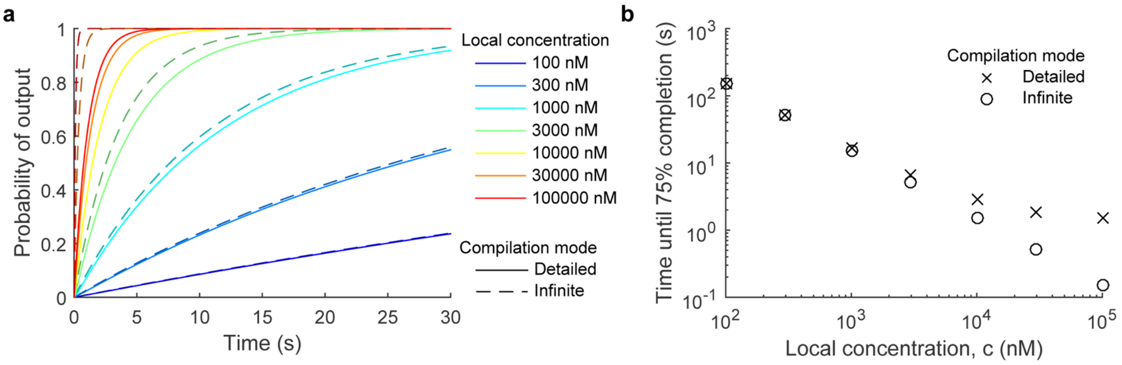


Figure 3. Analysis of a localized elementary strand displacement circuit for a range of local concentrations. (a) The transient probability of producing the circuit output was calculated in Visual DSD using probabilistic model checking for the models and kinetic parameters presented in Figure 2a (Detailed mode, dashed lines) and Figure 2b (Infinite mode, solid lines). The analysis was performed for a broad range of local concentrations. (b) Using the traces in panel a, the time at which the probability of output reached 0.75 was recorded as a proxy for the rate of signal propagation.

$$k_{\mathrm{eff}} = k_{\mathrm{bind}} \cdot \frac{k_{\mathrm{migrate}}}{k_{\mathrm{migrate}} + k_{\mathrm{unbind}}} \tag{1}$$

Using standard assumptions of mass action kinetics for well-mixed circuits, we simulate both CRNs as ordinary differential equations (ODEs), using a numerical solver built-in to Visual DSD (Figure 1a,b).

For all calculations in this article, unless stated otherwise, we assumed rate constants as defined in Figure 1, consistent with values reported previously.[26,27] The toehold binding rate k_{bind} has been estimated to vary between 6×10^{-3} and 4×10^{-4} nM^{-1} s^{-1}, depending on sequence composition of the toehold.[26,27] We therefore chose $k_{\mathrm{bind}} = 10^{-3}$ nM^{-1} s^{-1}, consistent with previous assumptions.[26] The toehold unbinding rate k_{unbind} has been estimated to vary over several orders of magnitude, depending primarily on toehold length but also on toehold sequence composition.[26,27] We therefore chose $k_{\mathrm{unbind}} = 10$ s^{-1}, consistent with a toehold length of 5 nucleotides.[26] Recent analysis of the branch migration rate[27] identified branch migration initiation as the rate-limiting step and proposed a value of approximately $k_{\mathrm{migrate}} = 1$ s^{-1}. We note that additional analysis can be readily performed using alternative rate constants.

A localized version of the elementary strand displacement device is shown in Figure 2, in which both the invading strand and the template are tethered to a DNA substrate. Elementary strands can be precisely positioned on a fully addressable DNA substrate by designing them as extensions to conventional substrate strands. In this setting, the topology of a localized DNA device is analogous to the topology of a digital electronic circuit. In contrast to the well-mixed circuit, the localized circuit contains only one copy of each strand and operates by localized interactions between spatially adjacent strands. Since there is only one copy of each strand, continuous deterministic simulation via mass action ODEs is inappropriate. It is well-known that, as the copy number of a discrete stochastic system decreases, mass action ODE approximations break down and stochastic simulation methods should be used instead. A standard approach involves the use of the stochastic simulation algorithm (SSA);[28] however, this produces a different output for each execution of the simulation. As a result, large numbers of simulations are needed in order to obtain an estimate of the expected behavior of the circuit. For an exact approach, we instead perform numerical integration of the chemical master equation (CME), which describes the distribution of all possible trajectories produced by SSA such that SSA samples the CME (see Methods for further description). Analysis of the CME is achieved by generating a continuous-time Markov chain (CTMC), which represents the full state space of the system, where each state corresponds to a vector of species populations and each position in the vector denotes a separate species. The primary reason that such approaches are not always used to analyze discrete stochastic systems is due to the large numbers of molecules involved, particularly in a well-mixed setting. For instance, if we consider a circuit with N species and 1000 copies of each, then the state space can approach 1000^N, which is intractable to enumerate even for small numbers of species, such as $N = 10$. In a localized setting, however, since there is only a single copy of each species, which can be either present or absent, we end up with a state space of less than 2^N, which is feasible to analyze for a much larger number of species (see Table S1 for the number of states in the circuits analyzed in this article). Furthermore, an important property of the localized circuit is that multiple copies of the circuit can be executed simultaneously in solution, with significantly reduced interference between circuits. This allows only a single copy of the circuit to be analyzed independently, in order to characterize the behavior of the population. Crucially, this renders localized circuits suitable for probabilistic model checking, which is usually not the case for well-mixed circuits.

To simulate the localized elementary strand displacement circuit (Figure 2), the Visual DSD tool was used to automatically generate a CTMC. We again consider two alternative CTMCs, a Detailed mode CTMC (Figure 2a), which represents toehold binding, unbinding, and branch migration as separate steps, and an Infinite mode CTMC (Figure 2b), which assumes that strand displacement occurs in a single step. To obtain the transition rates between states in the CTMC, we assume that the rate of interaction between two strands is scaled by the local concentration c of the strands, which takes into account the proximity of the strands to each other. This gives rise to a localized binding rate given by $c{\cdot}k_{\mathrm{bind}}$ for the Detailed mode CTMC and a localized displacement rate given by $c{\cdot}k_{\mathrm{eff}}$ for the Infinite mode CTMC. Here, we assume that the toehold unbinding rate k_{unbind} and the branch migration rate k_{migrate} are not scaled by the local concentration, although they may be affected by changes in free energy due to the tethering of strands to the substrate. For a given local concentration c, our probabilistic analysis of the CTMC uses

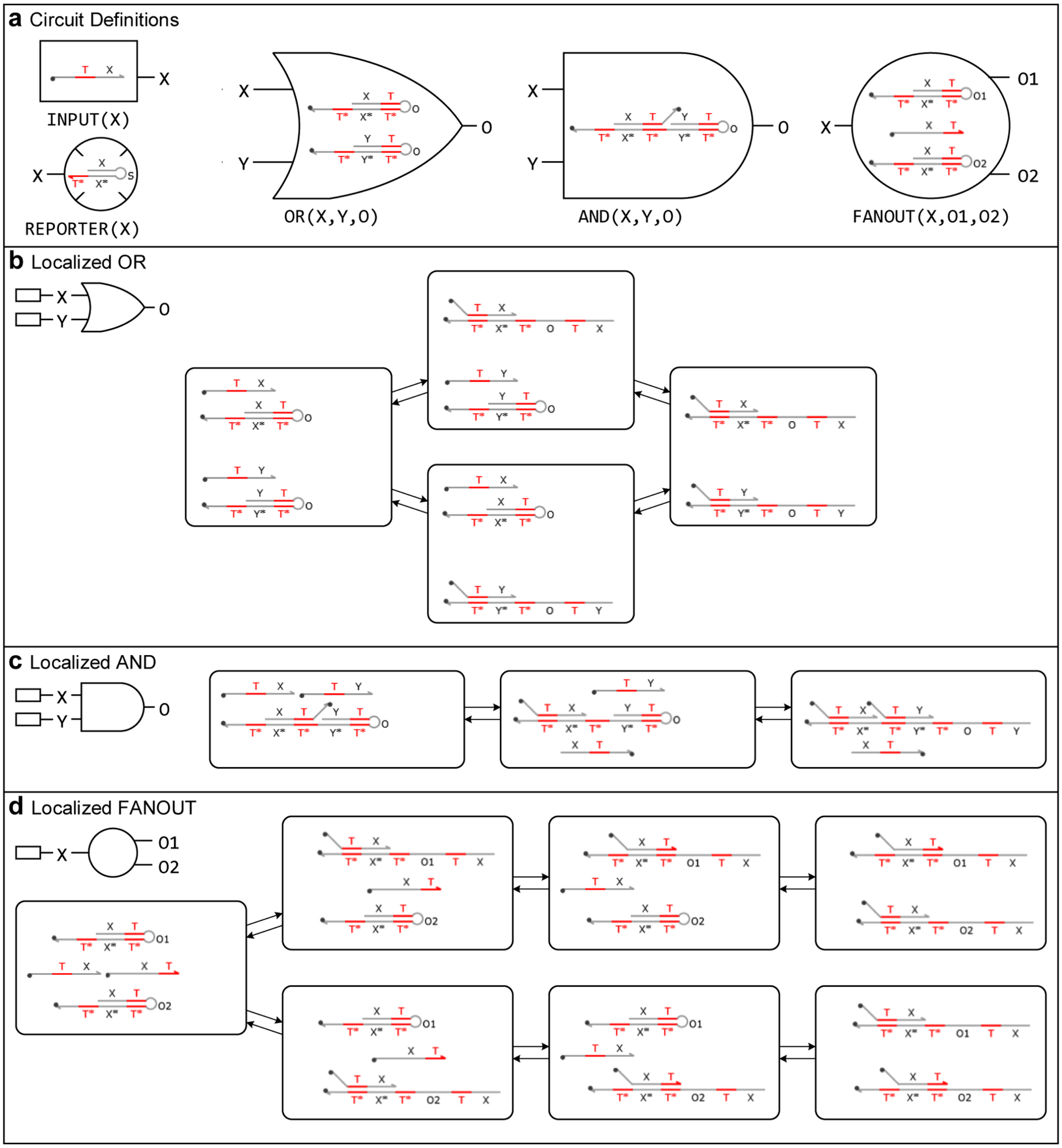


Figure 4. Localized designs for OR, AND, and FANOUT circuits. Tethers are represented by single strands with black dots at one end, where tether locations are approximate. (a) Summary of circuit designs. (b) The localized OR circuit computing the Boolean function $X \vee Y$ is implemented by two strands, $\langle XTOT^*X^*T^*\rangle$ and $\langle YTOT^*Y^*T^*\rangle$. The presence of either of the input strands, $\langle TX\rangle$ or $\langle TY\rangle$, triggers the exposure of the output TO, which is initially sequestered in a hairpin. The two different paths in the network represent the two possible orderings. A propagation gate enables the signal transduction $X \rightarrow O$ and is implemented by the first strand $\langle XTOT^*X^*T^*\rangle$ of the OR gate. (c) The localized AND circuit computing the Boolean function $X \wedge Y$ is implemented by a complex consisting of the strand $\langle YTOT^*Y^*T^*X^*T^*\rangle$ hybridized to the strand $\langle XT\rangle$. The presence of both input strands, $\langle TX\rangle$ and $\langle TY\rangle$, triggers the exposure of the output TO, which is initially sequestered in a hairpin. Note that the $\langle TX\rangle$ input binds first, followed by the $\langle TY\rangle$ input. (d) A localized degree two FANOUT gate transducing input signal X to two output signals O_1 and O_2. The localized input $\langle TX\rangle$ is present initially and can bind to one of the output gates to displace the first output. The initially present $\langle XT\rangle$ strand then displaces the input so that it can bind to the remaining output gate to displace the second output. The two different paths of the network represent the two possible orderings.

numerical integration of the CME, which becomes a linear system of ODEs that describe the probability of being in each state of the CTMC at a particular time. Details of the method are presented in the Supporting Information. The solution of the CME can then be queried to provide the mean and standard deviation of the copy number of each molecule, as depicted in Figure 2. To perform the simulation, we extended Visual DSD to permit efficient numerical integration of the

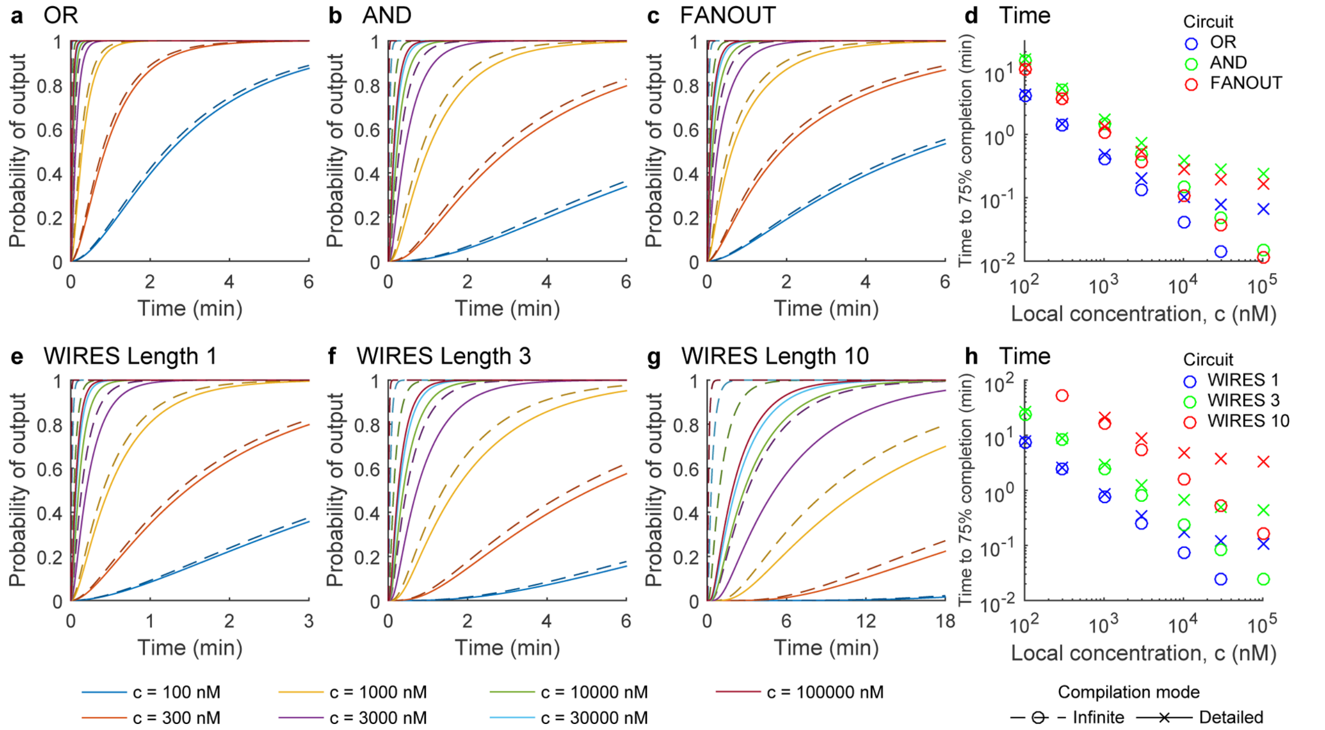


Figure 5. Analysis of simple fully localized circuits. Shown are probabilistic analyses of (a) OR ($X = 1, Y = 1$), (b) AND ($X = 1, Y = 1$), (c) FANOUT ($X = 1$), and (e−g) WIRES circuits, assuming all interactions are between tethered DNA molecules. In all panels, DSD models compiled in Infinite mode are compared against the same models compiled in Detailed mode. Rates were defined as in Figure 1. In each case, the effect of varying c is shown by lines of different colors, as indicated in the bottom legend. (d, h) The time at which the probability of completion reached 0.75 was recorded for each of the calculations in (d) panels a−c and (h) panels e−g. The crosses relate calculations for models compiled in Detailed mode, and the circles, for models compiled in Infinite mode.

CME, as described in the Methods and Supporting Information.

To illustrate the generality of the method with respect to specific biophysical parameters, we performed probabilistic analysis of the localized circuit (Figure 2) for a broad range of local concentrations (Figure 3), where different local concentrations are representative of different biophysical implementations. In Detailed mode, we scaled the binding rate k_{bind} by the local concentration c to obtain a localized binding rate, with the migration and unbinding rates assumed to be concentration-independent (Figure 2a). As the local concentration increased, toehold binding ceased to be the rate-limiting step and branch migration became limiting (Figure 3b). By considering the time at which the probability of output reaches 0.75, we observed a plateau in circuit speed as local concentration increased. In Infinite mode, we scaled the effective strand displacement rate k_{eff} by the local concentration c to obtain a localized strand displacement rate. The effective strand displacement rate k_{eff} is considered to be a good approximation when k_{bind} is low. However, this approximation begins to break down for high values of c and fails to account for the diminishing speedup observed in Detailed mode (Figure 3b). Therefore, for circuits with high local concentration, either alternative approximations for k_{eff} should be considered or Detailed mode should be used.

We emphasize that our method is parametrized by the local concentrations of the interacting strands, which can be determined in a variety of ways. One approach is to measure the fluorescence emitted by a bulk solution of localized circuits over time, and from this to infer the rate $c{\cdot}k_{\text{bind}}$,[7] from which an estimate of c can be obtained. This will rely on suitable estimates for k_{migrate} and k_{unbind}.[26] An alternative approach is to perform detailed molecular simulations of the localized strands, using biophysical methods such as OxDNA[29] to obtain estimates of localized transition rates. Another alternative is to compute the volumes explored by the two interacting strands[30] and to use these to provide an estimate of the local concentration. Even in the case where local concentrations are unknown, our method can still be used to simulate the behavior of the circuit for a broad range of local concentrations. This can provide insight into the local concentrations required for adequate circuit performance, which, in turn, can be used to inform the physical circuit implementation.

Design and Probabilistic Analysis of Elementary Localized Circuits. In this section, we propose localized hybridization designs for elementary Boolean logic circuits and use probabilistic analysis to characterize their performance as a function of local concentration. We consider spurious unintended leak interactions between strands and analyze the potential impact of leaks on circuit performance. We estimate a value of local concentration that maximizes the probability of computing the correct output over time in the presence of leaks. To determine how this concentration could be achieved in a physical implementation, we propose a biophysical model of tethered hybridization.

We start by designing elementary circuits that perform AND and OR Boolean logic, together with signal propagation and FANOUT (Figure 4a). These circuits are sufficient to implement more complex Boolean functions by means of dual rail logic, which uses one bit to encode a 0 and a separate bit to encode a 1. Gate behavior is implemented by cascades of toehold-mediated DNA strand displacement reactions, analo-

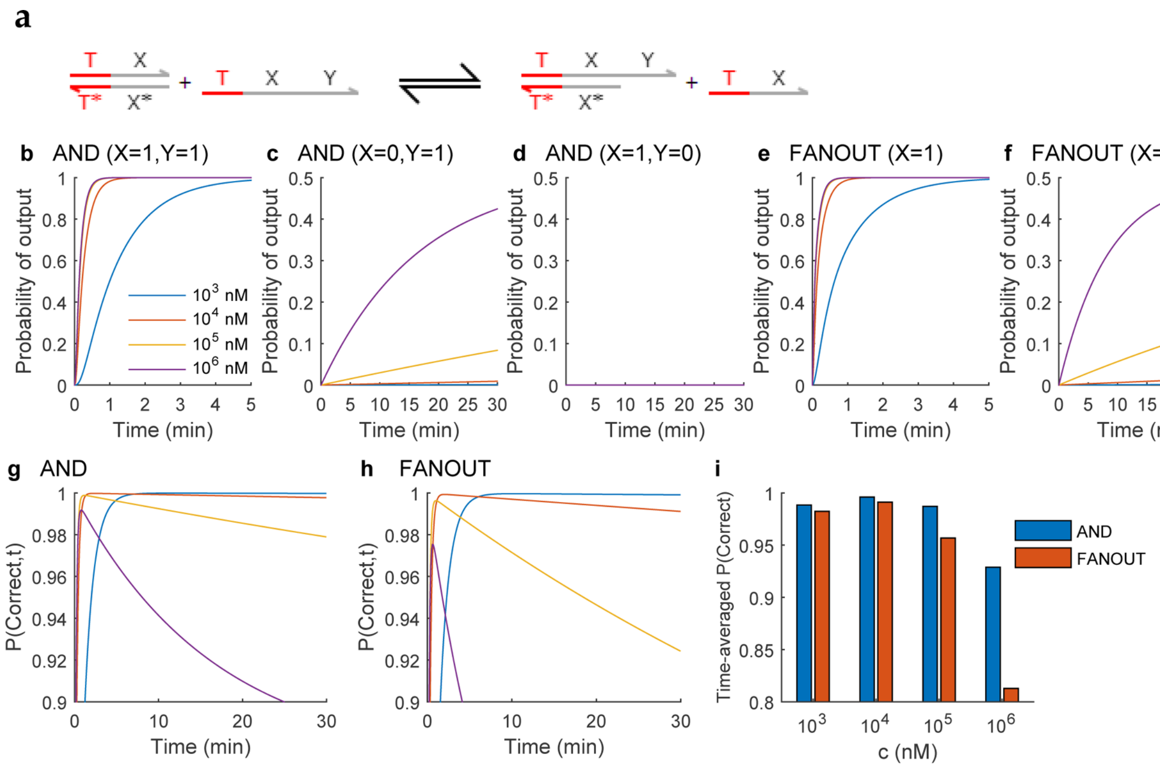
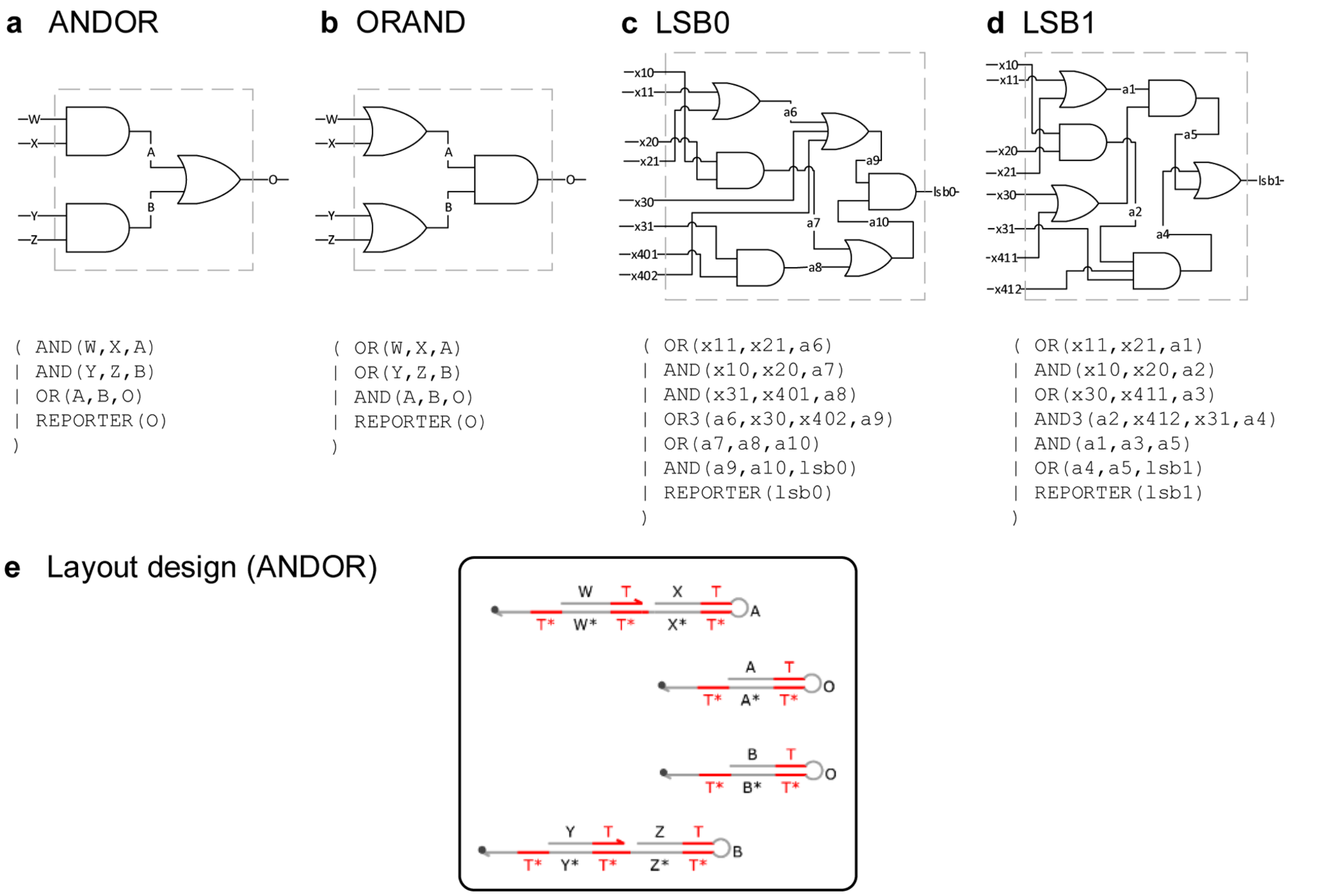


Figure 6. Leaks impact on circuit behavior only at high local concentrations. (a) Mechanism of zero toehold leaks. (b–f) Probabilistic analysis of DSD models compiled in Detailed mode with zero toehold leaks enabled at rate $c \times l$, where $l = 10^{-9}$ nM $^{-1}$ s^{-1} and $c = 10^4$ nM, with rates defined as in Figure 1. Inputs to AND and FANOUT circuits as indicated above each panel. (g) The time-dependent probability of correctness was calculated for the AND circuit as $^1/_4(P(O = 1,t|\mathrm{X} = 1,Y = 1) + P(O = 0,t|\mathrm{X} = 0,Y = 1) + P(O = 0,t|\mathrm{X} = 1,Y = 0) + P(O = 0,t|\mathrm{X} = 0,Y = 0))$. (h) The time-dependent probability of correctness was calculated for the FANOUT circuit as $^1/_2(P(O = 1,t|\mathrm{X} = 1) + P(O = 0,t|\mathrm{X} = 0))$. (i) The average correctness over the 30 min time interval was calculated for each circuit, illustrating an optimal choice of local concentration.

gous to the elementary reaction in Figure 1. Each strand displacement reaction is initiated via a universal toehold T, whose sequence is the same for all gates. The specificity of strand displacement is conferred by a set of specificity domains, W, X, Y, Z, which can differ between gates. OR logic can be achieved using two hairpin motifs (Figure 4b). For given input domains X and Y, we design strands with the sequences $\langle XTOT^*X^*T^*\rangle$ and $\langle YTOT^*Y^*T^*\rangle$, respectively, where O is the output domain that is initially concealed within a hairpin. In the presence of either of the input strands $\langle TX\rangle$ or $\langle TY\rangle$, one of the hairpins is opened to reveal the output O. A propagation gate, which acts as a wire that propagates signal, is achieved by simply using one of the two strands that make up the OR gate. By composing propagation gates in series, we can create signal transduction pathways between gates. AND logic can be implemented using a two-input AND gate consisting of a hairpin motif and a protector strand (Figure 4c). For input domains X and Y, the hairpin motif has sequence $\langle YTOT^*Y^*T^*X^*T^*\rangle$ and is hybridized to the protector strand $\langle XT\rangle$, where O is the output domain. In the presence of the first input strand $\langle TX\rangle$, the protector is displaced from the complex, exposing a toehold T^*. If the second input strand $\langle TY\rangle$ is also present, then it initiates strand displacement via this newly exposed toehold to open the hairpin and reveal the output O. A two-degree FANOUT gate is illustrated in Figure 4d. The signal X activates one of the two propagation gates, whose output regions O_1 and O_2 are assumed to be consumed by downstream gates. The fuel strand $\langle XT\rangle$ binds to the propagation gate using the newly exposed toehold T^* and kicks off the signal X, which can now activate the other propagation gate. Supporting higher numbers of inputs (fan-in) and outputs (fan-out) reduces the number of gates and simplifies circuit design. We also propose designs for fixed and limited degree fan-in and fan-out. A k degree fan-in OR gate can be achieved by using k hairpin motifs in parallel. A k degree fan-in AND gate can be achieved by using a hairpin motif with $k - 1$ protectors in series. The switching speed for this gate is inversely proportional to the degree of fan-in due to the serial nature of the gate. A k degree fan-out from a signal X can be implemented using k propagation gates transducing signal X to signals O_1,O_2, ..., O_k using $k - 1$ copies of a fuel strand $\langle XT\rangle$. Once again, the transduction speed of this gate is inversely propositional to the degree of fan-out due to the serial nature of the gate.

We applied our probabilistic model-checking analysis to the OR, AND, and FANOUT circuits, together with a WIRES circuit consisting of multiple signal propagation circuits in series. Specifically, we sought to determine how changing the local concentration affects the expected output of each circuit over time (Figure 5). As in Figure 3, we compared models compiled in Detailed mode with models compiled in Infinite mode, where the effective rate of strand displacement was set to $c{\cdot}k_{\mathrm{eff}}$. We found that the speed of circuits compiled in Detailed mode reached a plateau as local concentration was increased, since branch migration became limiting (Figure 5). The time at which a given circuit achieved 0.75 probability of output varied between circuits, proceeding as fast as 10 s for the one-step WIRE, compared with approximately 5 min for the 10-step WIRE (Figure 5h). In contrast, Infinite mode compilation revealed k_{eff} to be a poor approximation at high c, as the time to



```
( AND(W,X,A)
| AND(Y,Z,B)
| OR(A,B,O)
| REPORTER(O)
)
```

```
( OR(W,X,A)
| OR(Y,Z,B)
| AND(A,B,O)
| REPORTER(O)
)
```

```
( OR(x11,x21,a6)
| AND(x10,x20,a7)
| AND(x31,x401,a8)
| OR3(a6,x30,x402,a9)
| OR(a7,a8,a10)
| AND(a9,a10,lsb0)
| REPORTER(lsb0)
)
```

```
( OR(x11,x21,a1)
| AND(x10,x20,a2)
| OR(x30,x411,a3)
| AND3(a2,x412,x31,a4)
| AND(a1,a3,a5)
| OR(a4,a5,lsb1)
| REPORTER(lsb1)
)
```

Figure 7. Composition of elementary logic circuits to design complex circuits. Shown are graphical representations and module definitions in Visual DSD code for four circuits. (a) The ANDOR circuit computes the Boolean logic function $(W \wedge X) \vee (Y \wedge Z)$. (b) The ORAND circuit computes the Boolean logic function $(W \vee X) \wedge (Y \vee Z)$. (c, d) The LSB1 and LSB0 circuits report the least significant bit of the function $\lfloor \sqrt{x_4x_3x_2x_1} \rfloor$ where the x_i use dual rail logic in binary to encode integer numbers. For example, as 5 is 101 in binary, it is represented as $x_{11} = 1$, $x_{10} = 0$, $x_{21} = 0$, $x_{20} = 1$, $x_{31} = 1$, $x_{30} = 0$, $x_{41} = 0$, $x_{40} = 1$. Accordingly, the circuit output has LSB1 = 1 (LSB0 = 0) if the least significant bit of the square root function is 1 and LSB0 = 1 (LSB1 = 0) if the least significant bit is 0. The LSB0 circuit assumes that the x_{40} input has been forked prior to entering the LSB0 module and equivalently for the x_{41} input to the LSB1 circuit. (b) Example layout for an ANDOR circuit.

0.75 probability of output continued to increase linearly with local concentration. Therefore (as in Figure 3), for circuits with high local concentration, either alternative approximations for k_{eff} should be considered or Detailed mode should be used.

We considered spurious unintended leak interactions between strands to determine the potential impact of leaks on circuit performance. Leaks are spurious unprogrammed reactions that induce errors in computation. An important class of leaks, called zero toehold leaks, is illustrated in Figure 6a, where an invading strand displaces the incumbent strand even in the absence of an exposed toehold. These reactions proceed many orders of magnitude slower than toehold-mediated strand displacement reactions.

We analyzed the impact of zero toehold leaks on our elementary localized circuits using Visual DSD, which can automatically generate all zero toehold leaks. We compiled the circuits in Detailed mode with zero toehold leaks enabled. No leaks were observed for the OR and WIRES circuits; however, leaks were observed for the AND and FANOUT circuits (Figure 6b–e). The leak rate was set to $k_l = 10^{-9}$ nM^{-1} s^{-1}, which corresponds to a diffusion-based zero toehold leak rate reported previously.[5,26] For all localized reactions, we multiplied both the toehold binding rate and the leak rate by the local concentration c. For the AND circuit, we observed a difference between the Boolean inputs $(X,Y) = (0,1)$, which produced a low level of leak (Figure 6c), and the inputs $(X,Y) = (1,0)$, for which there was no observed leak (Figure 6d). This was due to the asymmetrical design of the AND gate with respect to the order of the inputs. As local concentration was increased, the probability of erroneous output for $(X,Y) = (0,1)$ increased dramatically (Figure 6c), while the speedup for $(X,Y) = (1,1)$ eventually reached a plateau due to the rate-limiting step of branch migration (Figure 6b). We observed a similar pattern for the FANOUT circuit, which leaked in the absence of input in a local concentration-dependent manner (Figure 6f). The FANOUT leak was due to the presence of the tethered fuel strand, which displaced the output even when no input was present. In general, we find that increasing c can simultaneously lead to a plateau in circuit speedup and an increase in circuit leak. Therefore, the design of localized circuits should not necessarily seek to increase the local concentration arbitrarily. Rather, optimization of signal-to-noise metrics is more appropriate.

To demonstrate this point, we calculated the probability of producing the correct output at a given time t. For the FANOUT circuit, this was defined as $P(\text{correct response}) = P(\text{output}|\text{input})\cdot P(\text{input}) + P(\text{no output}|\text{no input})\cdot P(\text{no input})$. We further assumed that the input is present or absent

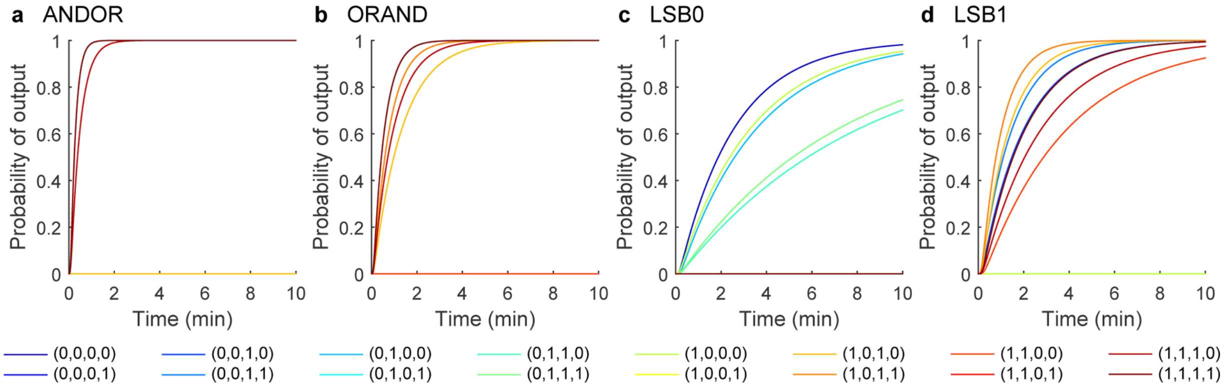


Figure 8. Analysis of complex fully localized circuits. Calculations of the probability of circuit completion for (a) ANDOR, (b) ORAND, (c) LSB0, and (d) LSB1 circuits, assuming all interactions are between tethered DNA molecules. All circuits are defined as in Figure 7. All inputs are assumed to be tethered to the origami. Models were compiled in detailed mode, and rates were defined as in Figure 1 but with $c = 10^4$ nM. For LSB0 and LSB1, a FANOUT gate was used to make two copies of the x40 and x41 inputs, respectively.

with equal probability (Figure 6g,h). This demonstrated that at earlier times the probability of a correct response was higher when the local concentration was higher, whereas at later times, the probability of a correct response was higher when the local concentration was lower. This can be explained by higher local concentrations producing correct outputs faster, although this comes at the cost of also producing incorrect leak outputs faster. By averaging over the 30 min interval considered here, we found that an optimal compromise was near $c = 10^4$ nM. Beyond this local concentration, the circuit speedup became limited by branch migration, whereas the leak speedup continued to increase.

To estimate how a specific value of local concentration could be achieved in a physical implementation, we propose a biophysical model of tethered hybridization and use it to provide estimates of intertether distance and signal length. Quantifying the local concentration is nontrivial due to the many factors that have the potential to influence this quantity. These include the distance between interacting molecules, the flexibility of DNA, and the lengths of the interacting molecules. Previously, a worm-like chain (WLC) model was used to describe the probability that the reactive domain of a DNA strand is in the vicinity of its downstream target.[30] By adapting this approach to consider the hairpin gate architecture in Figure 1, we used the WLC model to determine how the lengths of the signal strands and the intertether distances might affect the local concentration (Figure 12), similar to previous work.[30] This suggested experimentally feasible values of local concentration up to 10^6 nM. However, future experimental work is needed to determine whether such quantities are realizable in practice and also the extent to which additional biophysical factors may limit other aspects of signal propagation, such as the influence of the tethering geometry on branch migration rates, for instance, due to tension arising from tethered branch migration configurations.[31] More accurate characterization of the local concentration could be achieved using more detailed biophysical modeling approaches, such as OxDNA.[29]

Probabilistic Analysis of Complex Localized Circuits. In this section, we use our elementary localized circuits (Figure 4) to produce more complex ANDOR and ORAND circuits, together with a circuit that computes the square root of a four bit number. We use our probabilistic analysis method to prove the correctness of our designs with respect to their functional specifications and to analyze their performance over time.

We encoded an ANDOR circuit (Figure 7a), which implements the logic function $(W \wedge X) \vee (Y \wedge Z)$, and an ORAND circuit, which implements the logic function $(W \vee X) \wedge (Y \vee Z)$ (Figure 7b). We also encoded a set of circuits that together compute the square root of a four bit number (Figure 7c,d). We used a dual rail logic encoding, where each input bit was implemented by two strands representing the two distinct logical values that the bit can assume.[5] The result was reported over two output bits, the least significant bit (LSB) and the most significant bit (MSB), with two output strands for each bit. The LSB0 output is high only if the LSB is Boolean 0, and, conversely, the LSB1 output is high only if the LSB is Boolean 1. The LSB0 and LSB1 functions are given by LSB0 = $(x_{11} \vee x_{21} \vee x_{30} \vee x_{40}) \wedge ((x_{10} \wedge x_{20}) \vee (x_{31} \wedge x_{40}))$ and LSB1 = $(x_{41} \wedge x_{31} \wedge x_{10} \wedge x_{20}) \vee ((x_{11} \vee x_{21}) \wedge (x_{41} \vee x_{30}))$, where x_{i0} and x_{i1} are the 0 and 1 inputs, respectively, for the ith input bit (read from the lowest to the highest in significance). The Boolean formulas for MSB0 and MSB1 are given by MSB0 = $x_{30} \wedge x_{40}$ and MSB1 = $x_{31} \vee x_{41}$. Since these are identical to the AND and OR gates, respectively (Figure 4), the MSB circuits were not shown. The strategy for mapping Boolean functions to strand displacement gates that encode these circuits is summarized in Figure 7. Finally, to illustrate how a given circuit may be arranged on a DNA substrate, we have provided an example layout for the ANDOR circuit (Figure 7e). The AND gate $W \wedge X$ results in an intermediate A, which targets one component of the OR gate $A \vee B$. Similarly, the AND gate $Y \wedge Z$ results in an intermediate B, which targets the other component of the OR gate. More complex layouts will be required for larger circuits such as LSB, for which additional propagation circuits may be required in order to correctly route intermediate signals.

Our previous analysis of simple circuits considered a range of local concentrations. To simplify the presentation of results for the more complex circuits, we performed our analysis for a single local concentration. We used the concentration $c = 10^4$ nM, identified in the previous section, to optimize the signal-to-noise ratio of the circuits, although the same analysis could be performed with a different local concentration. We analyzed the circuits of Figure 7 to determine the expected output of each circuit over time, assuming localized inputs to all circuits. Since the LSB0 circuit assumes two separate copies of the x40 input, a

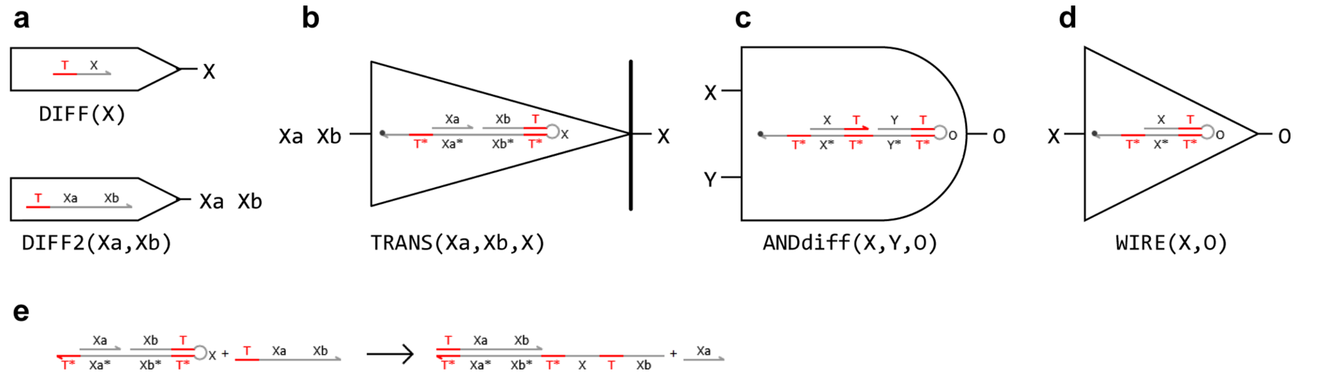


Figure 9. Diffusible inputs, transducer gate, and diffusible protector strands. (a) Diffusible inputs. (b) Transducer mechanism. Each localized input stand $\langle TX \rangle$ was replaced with a concentration of diffusible strands $\langle TX_aX_b \rangle$ in solution, together with a single localized transducer complex. (c) AND circuit with diffusible protector strands. (d) WIRES circuit. (e) Behavior of transducer.

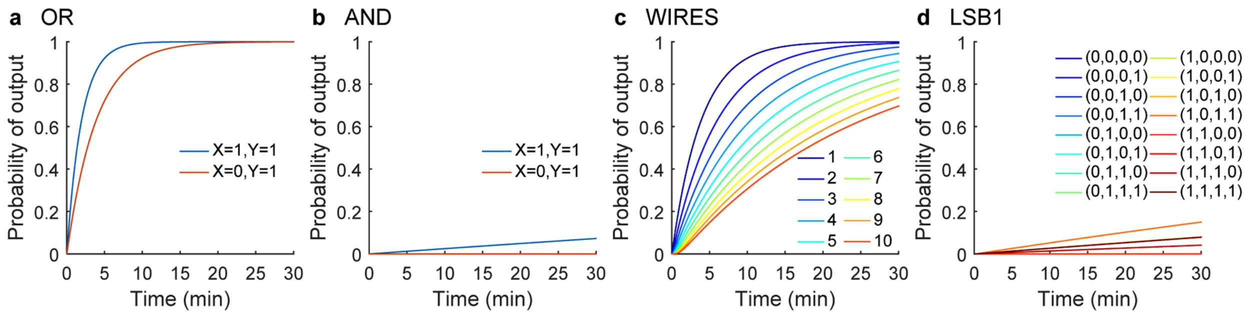


Figure 10. Analysis of localized circuits with diffusible inputs. Probability of circuit output over time, for circuits with diffusible inputs as defined in Figure S1a. All input strands were assumed to be diffusible and at constant concentration c_0 = 100 nM. Models were compiled in Detailed mode with rates as in Figure 1 and $c = 10^4$ nM. The presence of different inputs is indicated in the legends, with 1 corresponding to an input concentration of c_0. A dual rail logic encoding was used for the inputs to LSB1 (Figure 7d). Most of the circuits failed to reach 0.75 probability of completion within an hour.

FANOUT(x40,x401,x402) module was used. Similarly, the LSB1 circuit required the use of a FANOUT(x41,x411,x412) module. We found that all circuits produced the correct outputs for all combinations of Boolean inputs (Figure 8). In particular, the LSB0 output was high for Boolean inputs corresponding to 0 and 4–8, whereas the LSB1 output was high for Boolean inputs corresponding to 1–3 and 9–15. Most circuits reached above 75% probability of output within 10 min (Figure 8). There was some variability in the output time between different combinations of inputs due to the asymmetrical design of circuits with respect to the order of the inputs, specifically for the AND circuit. Note that this compares favorably with previous well-mixed designs,[5] where 75% completion was observed between 5 and 9 h for the LSB outputs. We emphasize that, since these circuits are spatially separated, they can be analyzed independently to provide an accurate characterization of performance when all circuits are executed simultaneously. This assumes a separate, localized set of inputs for each circuit, which can be achieved by using the FANOUT circuit of Figure 4.

Probabilistic Analysis of Localized Circuits with Diffusible Inputs. In this section, we consider the use of localized circuits in the presence of diffusible inputs, which corresponds to an important experimental context. We use our analysis to identify limitations in circuit performance in this context and propose a number of design improvements to overcome these limitations.

In an experimental setup, it is often convenient to trigger a localized circuit by means of diffusible inputs that can be added in solution. Since diffusible strands can be shared between circuits, we made the simplifying assumption that all diffusible inputs were present in excess at approximately constant concentration, given by the operating concentration c_0. This assumption allowed us to analyze the dynamics of each circuit independently. For testing purposes, we assumed c_0 = 100 nM; however, our analysis can be repeated for a broad range of concentrations. Assuming a nonlocalized binding rate k, the propensity of the reaction between a diffusible strand at constant concentration c_0 and a single tethered strand is given by $k \times c_0$ s^{-1}. For c_0 = 100 nM, this was considerably slower than the localized binding rate $k \times c$ when $c = 10^4$ nM. As a result, when all inputs were made diffusible (Figures 9a and S1a), we observed significant slow down in the completion times of most circuits (Figure 10). This slow down was due to the fact that all reactions in the circuits are reversible apart from the final reporter reaction. Since diffusion significantly decreased the binding rate of the inputs to the circuits, and the reverse reactions still took place at high local concentrations, the reverse reactions significantly slowed the forward progression of the circuits.

To counteract this problem, we considered two alternative strategies and compared their behaviors using probabilistic model checking. The first strategy was to untether the protector strands on all instances of the AND gates, resulting in a version of the gates with diffusible protector strands (Figure 9c). Removing the tether from these strands meant that they were no longer localized with high local concentration when released and therefore did not rebind at a fast rate. Instead, the protector

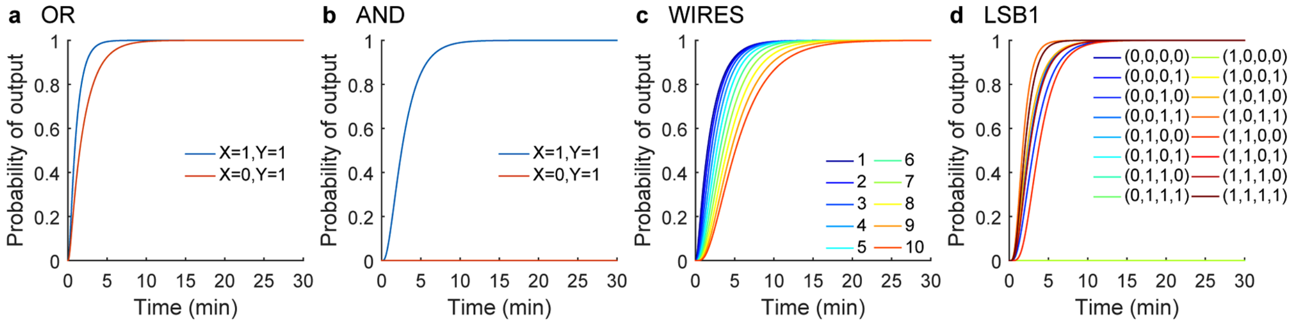


Figure 11. Analysis of localized circuits with transduced diffusible inputs and diffusible protector strands. Probability of circuit output over time, for circuits with transduced diffusible inputs as defined in Figure.S1b. All input strands were assumed to be diffusible and at constant concentration c_0 = 100 nM. In addition, all AND circuit protector strands of the form $\langle XT \rangle$ were assumed to be diffusible and at constant concentration c_1 = 10 nM, as defined in Figure 9c. Models were compiled in Detailed mode, and rates were defined as in Figure 1 with $c = 10^4$ nM. The presence of different inputs is indicated in the legends, with 1 corresponding to an input concentration of c_0. A dual rail logic encoding was used for the inputs to LSB1 (Figure 7d). All of the circuits reached 0.75 probability of completion within a few minutes.

strands diffused away from the substrate following displacement by the input. Applying this modification to all AND gates led to a dramatic speedup of all circuits utilizing these gates (Figure S2). In the second strategy, we used transducer gates to convert diffusible inputs to localized inputs with an irreversible step. To achieve this, for each input we used a concentration of diffusible strands of the form $\langle TX_aX_b \rangle$ in solution (Figure 9a), together with a single localized transducer complex (Figure 9b). This strategy resulted in even faster local computation than the use of diffusible protector strands (Figure S3). In addition, to overcome the spontaneous leak of the FANOUT circuit, we removed the FANOUT altogether and instead propagated all FANOUT signals to the diffusible inputs (Figure S1b). Propagation of the FANOUT was possible because the diffusible inputs were present in excess. Removing the FANOUT in this way effectively eliminated all leaks in the absence of inputs, allowing the circuit to be prepared and stored for long periods without leakage, prior to use. We note that for circuits with the FANOUT still included, leaks can be reduced by allowing the protector strand of the FANOUT to be untethered and instead added as diffusible fuel, allowing the circuit to be prepared and stored for long periods in the absence of both inputs and fuel. Finally, we combined all of these strategies, resulting in significantly faster circuits (Figure 11). To illustrate that the rate-limiting step was the transduction of diffusible inputs to localized inputs, we performed the same simulations but with $c_0 = 1\ \mu$M instead of 100 nM (Figure S4). We observed an order of magnitude speedup, solely due to the increase in operating concentration of the diffusible inputs. Thus, transducer circuits allow fast localized computation with diffusible inputs, and the concentration of these inputs can be increased in order to further speed up circuit computation. We further note that, for circuits with excess diffusible inputs, multiple localized circuits that require the same inputs can be executed simultaneously in solution. For instance, the LSB0, LSB1, MSB0, and MSB1 circuits can operate independently by sharing the diffusible the inputs x_{10}, x_{11}, x_{20}, x_{21}, x_{30}, x_{31}, x_{40}, x_{41}. As a result, the circuits can be analyzed separately to provide an accurate characterization of performance when all circuits are executed simultaneously.

Experimental Implementation Strategies. In this section, we propose a method for organizing DNA circuits on addressable DNA substrates by parallel synthesis that is experimentally feasible in principle and scales to a large number of gates. Although the methods mentioned in this section are yet to be experimentally demonstrated, they are extensions of highly successful techniques in DNA self-assembly.

We propose hierarchical assembly techniques[32] to organize our circuits on DNA lattices. Each DNA gate motif is designed as an extension of one of the strands that is part of the tile that assembles into a lattice. Since each tile in the finally formed lattice is uniquely addressable, the motifs can be precisely and specifically positioned on the lattice according to the circuit being implemented. We propose to organize our circuits on DNA origami.[33] Each DNA gate motif is an extension of a staple strand. Since the surface of the origami is uniquely addressable, the motifs can be precisely and specifically positioned on the origami. An inherent limitation to this approach is that DNA lattice formation is error-prone, such that, on average, staples are missing 7% of the time.[34] As such, a four gate cascade will form correctly only 75% of the time. This could be mitigated by including built-in redundancy in the circuit. In addition, since multiple copies of the localized circuit are executing in parallel in solution, the majority of circuits will produce the correct result. Consensus methods could be used to further improve robustness of the final output. More generally, further work is needed to increase reliability of assembly.

A key cause for concern is that when annealed the hairpin strands will interact with each other rather than folding up into the required hairpin motif. However, there is evidence that, when annealed, dilute (approximately nanomolar concentrations) interacting strands undergo unimolecular reactions and fold into hairpin motifs rather than hybridizing with each other via bimolecular reactions.[35] This is explained by noting that the hairpin structure is stable at a higher temperature than the intermolecular complex and that as the system is cooled the motifs form hairpins first and become kinetically trapped in the nonoptimal thermodynamic state. It is unclear if this assumption holds when the strands are locally concentrated, for instance, by tethering them close to each other. To avoid this problem, we design our motifs such that their hairpin structure is stable at higher temperatures than both the temperature at which they are stably incorporated into the substrate and the temperature at which the bimolecular complex is stable. When annealed, we expect the hairpin motif to form while the strands are dilute and not yet tethered to the substrate, and then the motifs are incorporated into the substrate.

Since the OR motifs are simply single-stranded hairpins, this is easily achieved by making the length of the specificity domain moderately longer than the length by which the motif is tethered to the origami. In practice, the tether length could be 16 nt while the length of the specificity domain could be 20 nt and the toehold domain could be 5 nt, making the stem of the OR hairpin motif 25 nt. The AND motif is slightly more problematic since it is a two-strand complex, a protector strand hybridized to a hairpin motif. By choosing lengths of 20 and 5 nt for the specificity and toehold domains, respectively, we can ensure that the protector—hairpin complex is stable at a higher temperature than the temperature at which the origami tether is stable. However, an upstream input to the AND motif would have similar stability with the AND hairpin as the protector—hairpin complex, since both are bimolecular reactions. This difficulty can be overcome in one of two ways. We can anneal the protector—hairpin complex separately, purify it, and then add it to the origami mix. While annealing the origami mix, we take care to not heat the sample above the melting temperature of the protector—hairpin complex. Alternatively, we can design the AND motif as a single hairpin motif and cleave the motif at the appropriate site after annealing by using a nicking enzyme. For this purpose, we can design the toehold sequence as the recognition domain of a nicking enzyme that cleaves one of the strands of a double helix upstream of its recognition site.[7] Note that a single nicking enzyme would be sufficient to prepare all protector—hairpin complexes and this process could be implemented in parallel. Also, the restriction enzyme would not nick the OR hairpin motifs, as the corresponding position in these structures is single-stranded.

A hierarchical method can be used to build more complex circuits. Small substrates can implement functional units that can be connected in a precise manner to synthesize computing architectures. A previously developed hierarchical assembly process[36] can be directly applied to build such circuits using tile-based assemblies. If origami is used as a substrate, then different origami can be connected to each other via sticky ends to form larger assemblies. One could also think of using a secondary scaffold to organize different origami in a precise manner to enable information flow between them. One simple layout for such architecture would be to have the computing elements in the middle of the origami and connect them up to neighboring origami via long signal transduction pathways that terminate at the edge of the origami. An advantage of using such architectures is the ability to plug and play various functional units. For example, if we have designed and experimentally tested a set of functional units, say an adder, subtracter, and square rooter, then we can build circuits that are composed of these functions by plugging these units into precise positions on the assembly. These functional units could be designed to ensure that they can communicate via the same signal transduction pathways for each input/output bit so that they can be composed seamlessly.

■ DISCUSSION

In this article, we presented a method for the probabilistic analysis of localized hybridization circuits. We proposed localized hybridization designs for elementary logic circuits, and in doing so, we developed a methodology for computing arbitrary Boolean functions, which can be used as a basis for future molecular integrated circuits. To illustrate the generality our method with respect to specific biophysical parameters, we performed probabilistic analysis of localized circuits for a broad range of local concentrations, where different local concentrations were representative of different biophysical implementations. We further considered spurious leak interactions between strands to determine their potential impact on circuit performance. We identified a trade-off between the speedup of circuit completion and the speedup of leak interactions and used this to determine an optimal value of local concentration. To determine how specific local concentrations could be implemented, we proposed a biophysical model of tethered hybridization and used it to estimate biophysical parameters of intertether distance and signal length corresponding to a given local concentration. We used our elementary localized circuits to produce more complex ANDOR and ORAND circuits, together with a circuit that computes the square root of a four bit number, which is a localized alternative to one of the most complex DNA strand displacement circuits realized to date.[5] We used our probabilistic analysis method to prove the correctness of our designs with respect to their functional specifications and to analyze their performance over time. Our square root circuit reached 75% completion in 5−9 min, compared with 5−9 h for a well-mixed design. We also considered the use of localized circuits in the presence of diffusible inputs, which corresponds to an important experimental context. We identified limitations in circuit performance in this context and proposed design improvements to overcome these limitations, resulting in performance comparable to that of fully localized circuits. Finally, we proposed a method for organizing DNA circuits on addressable DNA substrates by parallel synthesis that is experimentally feasible and scales to a large number of gates.

Localized molecular computing (MC) circuits have advantages over well-mixed circuits with respect to both speed and scalability. In local MC, molecules adjacent to one another on a substrate have a high local concentration, resulting in fast interactions. Furthermore, the interactions are preferentially sped up for the chosen set of colocalized molecules. In contrast, in global MC the speed of execution is limited by global rates of collision between molecules. These can be adjusted by increasing temperature; however, this is limited by the melting temperature of hybridization. They can also be adjusted by increasing molecular concentrations; however, since all concentrations are global, this also increases the rates of interference between all molecules. The speed of global MCs is therefore limited by practical considerations of temperature and concentration. In terms of scalability, since local MC involves interactions with only a fixed set of neighbors, this opens up possibilities of sequence reuse in the system. In particular, since local DNA hybridization circuits are spatially separated and cannot directly interact, we can reuse the same DNA sequences to build the same functionality on a different part of the system using very few distinct DNA sequences overall. Since localization results in reduced interactions between parallel components, this also means that localized circuits can be analyzed independently. Here, we exploit this parallelism to analyze the outputs of a square root circuit, consisting of four independent circuits that share only their diffusible inputs. Global MC on the other hand involves interactions among DNA strands that can be present anywhere in the reacting vessel, which implies a single global namespace for the sequences and hence considerably limits DNA sequence reusability.

The recent work on localized strand displacement circuits builds on steady experimental progress in the area of addressable substrates. A 4 × 4 fully addressable lattice[37] was built out of a cross tile[38] and later extended to an 8 × 8 fully addressable lattice[32] using hierarchical assembly techniques. The technique of DNA origami[33] uses a long scaffold DNA strand obtained from a viral genome, which folds into a desired shape by the use of hundreds of short synthetic DNA strands called staples. Origami has since been widely used as a substrate to arrange various molecules[39,40] and has been extended to form three-dimensional shapes.[41,42] We note that all designs based on tethering to an origami will be subjected to spatial constraints imposed by the length of the origami scaffold and that to assemble larger circuits on origami requires longer scaffolds. Naturally occurring long scaffolds might be problematic since they might exhibit strong secondary structure. This could be reduced by careful design of a synthetic scaffold. Long scaffolds might also interfere with the DNA sequences used for various gates, although this could be reduced by using spatial separation to rely on only a small set of DNA sequences.

As localized circuits inherently rely on low copy number interactions, their dynamics will be highly stochastic. As such, analyzing circuit completion times requires probabilistic methods. The major limitation of using probabilistic model checking is the lack of scalability to systems of many interacting components over multiple copy numbers, which can give rise to a combinatorial explosion in the state space of the system. However, localized circuits suffer less from this problem because their components are typically of abundance 0 or 1 (the molecule is present or absent at a given location) and therefore the state space is $\mathcal{O}(2^n)$ for n distinct species. Previously, it was shown how models of DNA hybridization circuits encoded in Visual DSD can be exported to PRISM software for making probabilistic queries.[24] Here, we improved on this workflow by implementing an alternative methodology for calculating transient probabilities. Our methodology uses direct numerical integration of the chemical master equation to provide the probability of being in any state at any time. While direct probabilistic analysis in Visual DSD has led to a major improvement in the efficiency of the PRISM-based workflow, it comes at the cost of a smaller set of logical queries being available for system analysis, as PRISM can analyze a large variety of temporal logic queries.

Our theoretical analysis of the effect of localization on DNA circuit speed up and varying severity of leaks is consistent with recent experimental work. Optimal speed ups were previously observed[10] at an intergate spacing of 20 nm. The effect of leaks was observed to increase at smaller intertether distances, which is consistent with our analysis. Our current models only account for zero toehold strand displacement leak reactions. A refined model could include other types of leaks, enabling a more detailed analysis of localized circuits. Additional errors in operation could include (i) leaks via spontaneous opening of the hairpin motifs, (ii) leaks via stacking induced strand displacement, and (iii) spurious toehold binding. Spontaneous opening of the hairpin motifs is likely to be rare under our operation conditions, since the stem of the hairpin is 25 bases. We refer to the end of the stem at the loop region as the head of the motif and the other end as its tail. Stacking-induced strand displacement is likely to occur via head-to tail-stacking of motifs; however, the loop region is likely to sterically hinder such stacking, destabilizing it. It would be interesting to experiment with carefully orienting the motifs on the origami surface such that these stackings strain the motif tether region and are hence sterically hindered. For instance, the motifs likely to undergo stacking could be oriented alongside each other. Since each motif has the same toehold binding region, the output of one motif may bind to the toehold region of a neighboring motif even if they are not designed to interact. This spurious interaction is prevented from setting off downstream reactions by the mismatch in the specificity domains. However, such reactions may block the toehold region and slow the operation of the circuit. This problem is present even with seesaw circuits[5] but does not seem to significantly affect its correct operation for moderate circuit size and sufficiently low concentrations. The spurious toehold interactions in our designs are restricted to the diameter of motifs reachable by the tethered motif, in contrast to the seesaw circuits where it is a global problem.

How localization impacts the kinetics of strand displacement remains mostly uncharacterized. In most strand displacement circuits in solution, the diffusion-limited formation of the encounter complex is rate-limiting. Therefore, we expect the primary consequence of localization to be an increase in k_{bind}, the on-rate of strand−gate binding. It is also possible that localization will affect k_{unbind} and k_{migrate}, although we do not consider that in this article. Future work is needed to better characterize these effects, ideally using methods that enable more fine-grained analysis at the nucleotide level, such as OxDNA.[29]

In this article, we considered local concentrations of 10^4 nM for the analysis of complex localized circuits. We compare this with prior experimental measurements of tethered circuits. The rate of signal transfer was previously reported[10] to be $k_{\text{trans}} = 0.0017\ \text{s}^{-1}$, whereas toehold binding was reported as $k_{\text{on}} = 5.7 \times 10^4\ \text{M}^{-1}\ \text{s}^{-1}$. Assuming that the rate of signal transfer is obtained by scaling signal binding by the local concentration ($k_{\text{trans}} = c \times k_{\text{on}}$), this corresponds to a local concentration c of approximately 30 nM. However, in this partially localized architecture, signal transfer was achieved via diffusion, which we expect to yield a much lower concentration than what might be possible with a fully localized architecture. The rate of propagation was previously estimated[14] to be $k_{\text{prop}} = 0.009\ \text{s}^{-1}$. Assuming rates as in Figure 1 with $k_{\text{prop}} = c \times k_{\text{eff}}$, this corresponds to a local concentration c of approximately 100 nM. However, signal propagation also relied on a diffusible nicking enzyme prior to the localized reaction between two tethered strands. Therefore, the 0.009 s^{-1} rate is itself an approximation of a multistep process, involving diffusible enzyme nicking followed by toehold binding, and it is difficult to quantify the rates of these individual steps from the measurements published thus far. In the hairpin-based architecture of Figure 1, there is no reliance on diffusible enzyme nicking and therefore the rate of signal propagation is likely to be significantly faster than 0.009 s^{-1}, corresponding to a significantly higher local concentration. More generally, further work is needed to more accurately characterize hairpin binding rates, based on experimental measurements.[43,44]

An extended abstract of this work was previously presented,[8] which we have significantly extended as follows. While previous work[8] considered only simulation of localized circuits, here we propose a method for probabilistic model checking, which we implement within the Visual DSD design tool and use to perform extensive circuit analysis. Probabilistic model checking eliminates the need to average over large numbers of stochastic simulations, making it more reliable and efficient for systems

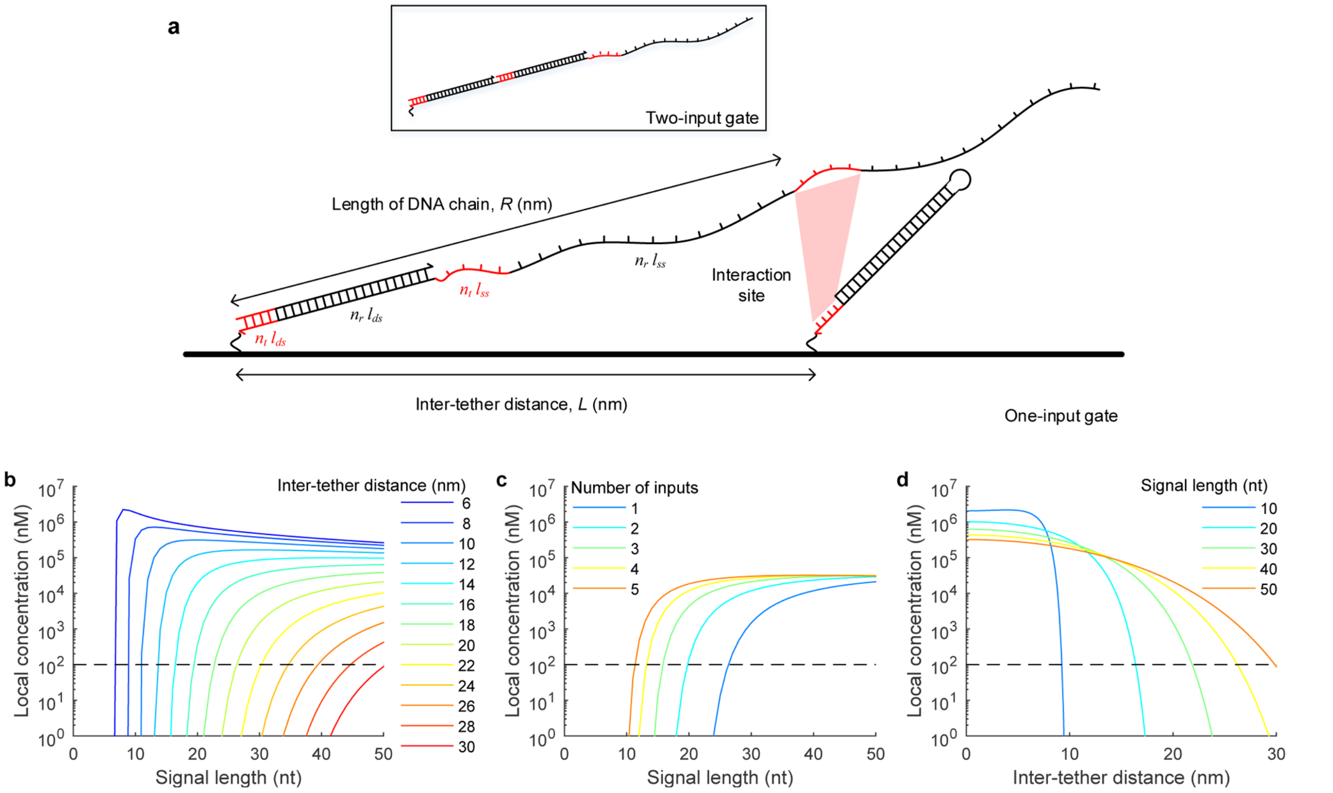


Figure 12. A biophysical model of tethered hybridization. (a) A graphical depiction of the interaction between an open tethered gate and a nearby (closed) target. Indicated are double- and single-stranded sections on the open tethered gate, which influence its flexibility and thus the local concentration of the interaction. The boxed inset illustrates an open tethered gate that has received two inputs, resulting in a longer double-stranded section. (b−d) Using a worm-like chain (WLC) model, the local concentration is calculated approximately as a function of the number of nucleotides in a signal strand, which defines the length of both the double-stranded portion of the open tethered gate and single-stranded portion upstream of the interaction site. In (b) and (d), a single input strand is assumed and therefore the double-stranded portion is one signal strand long, whereas in (c), the number of input strands to the open tethered gate scales the double-stranded portion. In (c), an intertether distance of 20 nm is assumed.

with moderately small state spaces (up to ≈10^6 states). We were able to analyze all circuits in Detailed mode without leaks (Figures 5, 8, and 11) and the elementary circuits with leaks (Figure 6). By using Detailed mode, we were able to observe the initiation of branch migration becoming a limiting factor to localized circuit progression. On the basis of our analysis, we propose a number of strategies for circuit optimization to reduce the effects of leaks. Finally, as an alternative to a rigid biophysical model,[8] we used an approach based on a worm-like chain model,[30] which provides the potential for a more accurate estimate of local concentration through simple calculations. However, this model remains to be parametrized against experimental measurements.

The major focus of this study has been to demonstrate how probabilistic analysis can be applied to localized molecular circuit implementation strategies. Future work will also involve combining the present approach with recent extensions to Visual DSD for abstract modeling of tethered circuits,[45] in which more fine-grained modeling of local concentrations between species can be achieved. While a specific architecture has been illustrated, our approach is also generalizable to other architectures, such as a localized hairpin architecture that relies on the binding of a diffusible fuel molecule.[9] This can be modeled in Visual DSD in a similar fashion, by assuming a constant population of fuel. In the future, additional work will be required to test the validity of the biophysical model proposed herein. By collecting data for a range of circuits and comparing with model analysis, it should be possible to infer values of local concentration using previously established methods.[7,26] By developing a computational framework for the probabilistic analysis of localized DNA hybridization circuits, we have enabled a principled design of circuits that can identify constraints on kinetic parameters to optimize circuit performance, which, in turn, can be used to guide subsequent biophysical implementations.

■ METHODS

Biophysical Approximation of the Effect of Localization on the Kinetics of DNA Hairpin Interactions. To determine the potential speedup of localized hybridization circuits, we applied a biophysical model of tethered hybridization[30] to quantify a local concentration c that depended on the parameters of localized design. In Supporting Information section 1, we present a formal description and derivation of the speedup in the toehold binding rate due to the effective local concentration of the incoming strand. We first calculate an approximate local concentration c for the incoming strand and calculate the new toehold binding rate $\tilde{k}$ as the product of the original toehold binding rate k and the local concentration c. We assume that the gates are tethered to their substrate via a short single strand of DNA that has negligible persistence length and hence acts as a completely flexible hinge. Once input strands have bound the gate and opened up the hairpin loop, the gate will have a double-stranded region closest to the tether, connected to a single-stranded region (Figure 12a). To approximate the local concentration for this regime, we assume that the open hairpin acts as a worm-like chain (WLC),[30] with persistence length 2 nm and total length defined as

$$R = N_{\mathrm{I}}(n_{\mathrm{t}} + n_{\mathrm{r}})l_{\mathrm{ds}} + (n_{\mathrm{t}} + n_{\mathrm{r}})l_{\mathrm{ss}}$$

where n_{t} is the number of nucleotides in a toehold domain, n_{r} is the number of nucleotides in a recognition domain, l_{ds} is the internucleotide distance for double-stranded DNA (0.34 nm), and l_{ss} is the internucleotide distance for single-stranded DNA (Figure 12a). The first term is multiplied by the number of inputs to the open gate, N_{I}, as this will extend the double-stranded region accordingly. By using this formula to define the length of an open hairpin molecule, we are making the simplifying assumption that double-stranded DNA is as flexible as single-stranded DNA. As we expect double-stranded DNA to be more rigid, the local concentrations calculated using this method are conservative lower bounds.

To illustrate how physical parameters modulate the potential speedup of localized hybridization circuits, we calculated the local concentration over a variety of scenarios. As expected, the shorter the distance between tethers, the higher the speedup predicted by the WLC model (Figure 12b). For experimentally realizable distances of between 6 and 20 nm,[10,14] this model predicts the local concentration to span several orders of magnitude as a function of the length of the signal (Figure 12). However, for a specified intertether distance, we found that there was an optimal length for the open hairpin, such that local concentration was maximized. The optimal signal length increased as intertether distance increased, as expected. For 12 nm tether spacing, 29 nucleotides was optimal, whereas for 14 nm tether spacings, 42 nucleotides was optimal. As increasing the number of inputs to a gate increases the length of the open hairpin, shorter signals are required for equivalent increases in local concentration (Figure 12c reports predictions for an intertether distance of 20 nm). The optimal signal length, therefore, decreased with additional inputs. Finally, we found that, for a given signal length, increasing the intertether distance generally decreased the local concentration (Figure 12d). Taken together, this analysis indicates that the selection of geometry and nucleotide sequence can have important consequences for the effect of localization on the speed of signal propagation.

Probabilistic Analysis of Localized DNA Circuits. Tethered hybridization circuits will naturally give rise to a large degree of stochasticity due to the low copy number variations inherent in each interaction on the origami. To analyze these stochastic dynamics, we extended Visual DSD with new functionality for probabilistic model checking. Specifically, we use the chemical reaction network produced from the compilation of DSD models to derive a continuous-time Markov chain (CTMC), which can be queried for probabilistic properties such as the probability of the system reaching a given state within a given time *t*. Model checking in this context refers to the rigorous analysis of stochastic processes. In well-mixed chemical systems (no localization), the probability that there are a given number of copies of each molecule type at a particular time can also be described by the chemical master equation (see Supporting Information for description), but the high molecule counts of each interacting molecule gives rise to a combinatorial explosion in the number of states in the CTMC. Here, since we are not using a well-mixed system, the localized stochastic kinetics transforms our state space into vectors where simply the presence or absence of a molecule at a specific site is tracked. Furthermore, the number of different molecules that may exist at a specific site is strongly limited by the topology of the localized circuit. Therefore, the associated combinatorics become very restricted, and the state space becomes tractable to analyze. Since each circuit operates largely independently, it is sufficient to analyze only a single circuit in order to obtain predictions about any number of circuits running in parallel. In particular, the average behavior of all circuits can be calculated as the stochastic mean of the master equation, which, in this case, may deviate significantly from ODE representations of the system due to stochastic effects inherent at low copy numbers. Additionally, since each tethered molecule will be at most of copy number 1, the stochastic mean is precisely equal to the probability of it being present. Therefore, using integration of the CME, it is convenient to track the probability of completion of any given localized circuit, i.e., the probability that the output strand exists at a specific location at a given time *t*. In order to calculate this, we summed the probabilities $P(x,t)$ for which x has a 1 in the relevant position.

■ ASSOCIATED CONTENT

Ⓢ Supporting Information

(1) Derivation of the local concentration, (2) example of the chemical master equation, (3) implementation details and usage of Visual DSD, (4) supplementary analysis plots, (5) description of the generation of CTMCs from Visual DSD, and (6) DSD code for all models. The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acssynbio.5b00044.

■ AUTHOR INFORMATION

Corresponding Authors

*(A.P.) E-mail: andrew.phillips@microsoft.com.
*(J.R.) E-mail: reif@cs.duke.edu.

Present Address

#Harish Chandran, Google Inc., 1600 Amphitheater Parkway, Mountain View, CA 94043, USA

Author Contributions

⊥N.D. and H.C. contributed equally to this work.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

We thank Erik Winfree, David Zhang, and Bernard Yurke for useful discussions on the localized strand displacement process and Anthony Genot for assistance with the worm-like chain model. We thank anonymous referees for pointing out relevant prior work and critical suggestions on modeling of tethered systems. We also thank Sudhanshu Garg for assistance in creating figures for a preliminary draft and Archana Ramamoorthy for proof-reading. This work was supported by NSF EMT grant nos. CCF-0829797 and CCF-0829798.

■ REFERENCES

(1) Zhang, D., and Seelig, G. (2011) Dynamic DNA Nanotechnology using Strand Displacement Reactions. *Nat. Chem. 3*, 103−113.

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

(3) Yin, P., Choi, H., Calvert, C., and Pierce, N. (2008) Programming Biomolecular Self-assembly Pathways. *Nature 451*, 318−322.

(4) Qian, L., and Winfree, E. (2011) A Simple DNA Gate Motif for Synthesizing Large-scale Circuits. *J. R. Soc., Interface 8*, 1281.

(5) Qian, L., and Winfree, E. (2011) Scaling up Digital Circuit Computation with DNA Strand Displacement Cascades. *Science 332*, 1196−1201.

(6) Qian, L., Winfree, E., and Bruck, J. (2011) Neural Network Computation with DNA Strand Displacement Cascades. *Nature 475*, 368−372.

(7) Chen, Y.-J., Dalchau, N., Srinivas, N., Phillips, A., Cardelli, L., Soloveichik, D., and Seelig, G. (2013) Programmable chemical controllers made from DNA. *Nat. Nanotechnol. 8*, 755−762.

(8) Chandran, H., Gopalkrishnan, N., Phillips, A., and Reif, J. H. (2011) Localized Hybridization Circuits. *Lect. Notes Comput. Sci. 6937*, 64−83.

(9) Muscat, R. A., Strauss, K., Ceze, L., and Seelig, G. (2013) *DNA-based molecular architecture with spatially localized components*, 40th Annual International Symposium on Computer Architecture, Tel-Aviv, Israel, June 23−27, pp 177−188.

(10) Teichmann, M., Kopperger, E., and Simmel, F. C. (2014) Robustness of Localized DNA Strand Displacement Cascades. *ACS Nano 8*, 8487−8496.

(11) Green, S., Bath, J., and Turberfield, A. (2008) Coordinated Chemomechanical Cycles: A Mechanism for Autonomous Molecular Motion. *Phys. Rev. Lett. 101*, 238101.

(12) Omabegho, T., Sha, R., and Seeman, N. C. (2009) A bipedal DNA Brownian motor with coordinated legs. *Science 324*, 67−71.

(13) Lund, K., Manzo, A. J., Dabby, N., Michelotti, N., Johnson-Buck, A., Nangreave, J., Taylor, S., Pei, R., Stojanovic, M. N., Walter, N. G., Winfree, E., and Yan, H. (2010) Molecular robots guided by prescriptive landscapes. *Nature 465*, 206−210.

(14) Wickham, S. F. J., Endo, M., Katsuda, Y., Hidaka, K., Bath, J., Sugiyama, H., and Turberfield, A. J. (2011) Direct observation of stepwise movement of a synthetic molecular transporter. *Nat. Nanotechnol. 6*, 166−169.

(15) Wickham, S. F. J., Bath, J., Katsuda, Y., Endo, M., Hidaka, K., Sugiyama, H., and Turberfield, A. J. (2012) A DNA-based molecular motor that can navigate a network of tracks. *Nat. Nanotechnol. 7*, 169−173.

(16) Liber, M., Tomov, T. E., Tsukanov, R., Berger, Y., and Nir, E. (2015) A Bipedal DNA Motor that Travels Back and Forth between Two DNA Origami Tiles. *Small 11*, 568−575.

(17) Scheible, M. B., Pardatscher, G., Kuzyk, A., and Simmel, F. C. (2014) Single molecule characterization of DNA binding and strand displacement reactions on lithographic DNA origami microarrays. *Nano Lett. 14*, 1627−1633.

(18) Wong, W., and Scott, J. D. (2004) AKAP signalling complexes: focal points in space and time. *Nat. Rev. Mol. Cell Biol. 5*, 959−970.

(19) Lakin, M. R., Youssef, S., Polo, F., Emmott, S., and Phillips, A. (2011) Visual DSD: a design and analysis tool for DNA strand displacement systems. *Bioinformatics 27*, 3211−3213.

(20) Lakin, M. R., Youssef, S., Cardelli, L., and Phillips, A. (2012) Abstractions for DNA circuit design. *J. R. Soc., Interface 9*, 470−486.

(21) Amir, Y., Ben-Ishay, E., Levner, D., Ittah, S., Abu-Horowitz, A., and Bachelet, I. (2014) Universal computing by DNA origami robots in a living animal. *Nat. Nanotechnol. 9*, 353−357.

(22) Kwiatkowska, M., Norman, G., and Parker, D. (2011) PRISM 4.0: Verification of Probabilistic Real-Time Systems. *Lect. Notes Comput. Sci. 6806*, 585−591.

(23) Dannenberg, F., Kwiatkowska, M., Thachuk, C., and Turberfield, A. J. (2013) DNA Computing and Molecular Programming: 19th International Conference, DNA 19, Tempe, AZ, September 22−27, pp 31−45.

(24) Lakin, M. R., Parker, D., Cardelli, L., Kwiatkowska, M., and Phillips, A. (2012) Design and analysis of DNA strand displacement devices using probabilistic model checking. *J. R. Soc., Interface 9*, 1470−1485.

(25) Reibman, A., and Trivedi, K. (1988) Numerical transient analysis of Markov models. *Computers & Operations Research 15*, 19−36.

(26) Zhang, D. Y., and Winfree, E. (2009) Control of DNA Strand Displacement Kinetics Using Toehold Exchange. *J. Am. Chem. Soc. 131*, 17303−17314.

(27) Srinivas, N., Ouldridge, T. E., Sulc, P., Schaeffer, J. M., Yurke, B., Louis, A. A., Doye, J. P. K., and Winfree, E. (2013) On the biophysics and kinetics of toehold-mediated DNA strand displacement. *Nucleic Acids Res. 41*, 10641−10658.

(28) Gillespie, D. (1977) Exact Stochastic Simulation of Coupled Chemical Reactions. *J. Phys. Chem. 81*, 2340−2361.

(29) Ouldridge, T. E., Louis, A. A., and Doye, J. P. K. (2011) Structural, mechanical, and thermodynamic properties of a coarse-grained DNA model. *J. Chem. Phys. 134*, 085101.

(30) Genot, A., Zhang, D., Bath, J., and Turberfield, A. (2011) Remote Toehold: A Mechanism for Flexible Control of DNA Hybridization Kinetics. *J. Am. Chem. Soc. 133*, 2177−2182.

(31) Šulc, P., Ouldridge, T. E., Romano, F., Doye, J. P. K., and Louis, A. A. (2014) Simulating a burnt-bridges DNA motor with a coarse-grained DNA model. *Nat. Comput. 13*, 535−547.

(32) Dwyer, C., and Pistol, C. (2007) Scalable, Low-cost, Hierarchical Assembly of Programmable DNA Nanostructures. *Nanotechnology 18*, 125305−125309.

(33) Rothemund, P. (2006) Folding DNA to Create Nanoscale Shapes and Patterns. *Nature 440*, 297−302.

(34) Jungmann, R., Steinhauer, C., Scheible, M., Kuzyk, A., Tinnefeld, P., and Simmel, F. C. (2010) Single-molecule kinetics and super-resolution microscopy by fluorescence imaging of transient binding on DNA origami. *Nano Lett. 10*, 4756−4761.

(35) Dirks, R., Bois, J., Schaeffer, J., Winfree, E., and Pierce, N. (2007) Thermodynamic Analysis of Interacting Nucleic Acid Strands. *SIAM Rev. 49*, 65−88.

(36) Park, S. H., Pistol, C., Ahn, S. J., Reif, J., Lebeck, A., and LaBean, C. D. T. (2006) Finite-Size, Fully Addressable DNA Tile Lattices Formed by Hierarchical Assembly Procedures. *Angew. Chem., Int. Ed. 45*, 735−739.

(37) Park, S.-H., Yin, P., Liu, Y., Reif, J., LaBean, T., and Yan, H. (2005) Programmable DNA Self-assemblies for Nanoscale Organization of Ligands and Proteins. *Nano Lett. 5*, 729−733.

(38) Yan, H., Park, S. H., Finkelstein, G., Reif, J., and LaBean, T. (2003) DNA-Templated Self-Assembly of Protein Arrays and Highly Conductive Nanowires. *Science 301*, 1882−1884.

(39) Gu, H., Chao, J., Xiao, S.-J., and Seeman, N. (2010) A Proximity-based Programmable DNA Nanoscale Assembly Line. *Nature 465*, 202−205.

(40) Lin, C., Liu, Y., and Yan, H. (2007) Self-Assembled Combinatorial Encoding Nanoarrays for Multiplexed Biosensing. *Nano Lett. 7*, 507−512.

(41) Douglas, S., Dietz, H., Liedl, T., Hogberg, B., Graf, F., and Shih, W. (2009) Self-assembly of DNA into Nanoscale Three-dimensional Shapes. *Nature 459*, 414−418.

(42) Dietz, H., Douglas, S., and Shih, W. (2009) Folding DNA into Twisted and Curved Nanoscale Shapes. *Science 325*, 725−730.

(43) Tsukanov, R., Tomov, T. E., Berger, Y., Liber, M., and Nir, E. (2013) Conformational dynamics of DNA hairpins at millisecond resolution obtained from analysis of single-molecule FRET histograms. *J. Phys. Chem. B 117*, 16105−16109.

(44) Tsukanov, R., Tomov, T. E., Masoud, R., Drory, H., Plavner, N., Liber, M., and Nir, E. (2013) Detailed study of DNA hairpin dynamics using single-molecule fluorescence assisted by DNA origami. *J. Phys. Chem. B 117*, 11932−11942.

(45) Lakin, M. R., Petersen, R., Gray, K. E., and Phillips, A. (2014) *Abstract Modelling of Tethered DNA Circuits*, DNA Computing and Molecular Programming: 20th International Conference, DNA 20, Kyoto, Japan, September 22−26, pp 132−147.