




Geometric Enumeration of Localized DNA Strand Displacement Reaction Networks

Matthew R. Lakin¹   

Department of Computer Science, University of New Mexico, USA

Department of Chemical & Biological Engineering, University of New Mexico, USA

Center for Biomedical Engineering, University of New Mexico, USA

Sarika Kumar

Department of Computer Science, University of New Mexico, USA

Abstract

Localized molecular devices are a powerful tool for engineering complex information-processing circuits and molecular robots. Their practical advantages include speed and scalability of interactions between components tethered near to each other on an underlying nanostructure, and the ability to restrict interactions between more distant components. The latter is a critical feature that must be factored into computational tools for the design and simulation of localized molecular devices: unlike in solution-phase systems, the geometries of molecular interactions must be accounted for when attempting to determine the network of possible reactions in a tethered molecular system. This work aims to address that challenge by integrating, for the first time, automated approaches to analysis of molecular geometry with reaction enumeration algorithms for DNA strand displacement reaction networks that can be applied to tethered molecular systems. By adapting a simple approach to solving the biophysical constraints inherent in molecular interactions to be applicable to tethered systems, we produce a localized reaction enumeration system that enhances previous approaches to reaction enumeration in tethered system by not requiring users to explicitly specify the subsets of components that are capable of interacting. This greatly simplifies the user's task and could also be used as the basis of future systems for automated placement or routing of signal-transmission and logical processing in molecular devices. We apply this system to several published example systems from the literature, including both tethered molecular logic systems and molecular robots.

2012 ACM Subject Classification Computer systems organization → Molecular computing

Keywords and phrases Localized circuits, reaction enumeration, DNA strand displacement, geometry, molecular computing

Digital Object Identifier 10.4230/LIPIcs.DNA.30.1

Supplementary Material *Software*: <https://github.com/matthewlakin/LocalizedEnumerator>

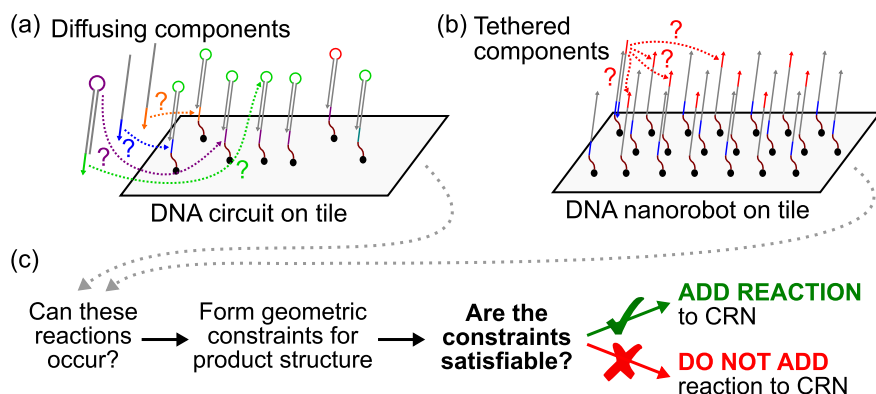
Funding This material is based upon work supported by the National Science Foundation under grants 1518861, 1814906, and 2044838.

1 Introduction

Recent advances in DNA nanotechnology, in particular the advent of DNA origami [23], have enabled spatially organized molecular systems to be designed and built to perform sophisticated tasks at the nanoscale using components tethered at particular locations on DNA origami tile surface. Notable examples have included cargo sorting robots [29] and signal transmission lines and Boolean logic circuits [6, 4, 5]. Both of these examples were implemented using toehold-mediated DNA strand displacement reactions [35, 26]. In addition to the practical applications of directed motion of cargo across the surface, such tethered

¹ Corresponding author





■ **Figure 1** Geometric reaction enumeration for systems involving tethered components. (a) Example of diffusing input strands and fuel hairpins interacting with localized components of a DNA logic circuit that are tethered to a DNA origami tile [6]. (b) Example of a fully localized DNA nanorobot walking between multiple track locations, all tethered to a DNA origami tile [28]. (c) Workflow for calculating whether to admit a candidate reaction based on molecular geometry.

systems have partial advantages for molecular computing, including enhanced computational speed, given the fast kinetic rates of localized interactions, and scalability, given the ability to reuse identical sequences to construct similar components in different parts of a spatially organized circuit. Furthermore, a larger DNA nanostructure could be used as a carrier to deliver multiple components from a multi-part diagnostic or therapeutic molecular circuit, attached to that one nanostructure, to targeted cells or tissues for biomedical applications. This could avoid potential issues with mismatched stoichiometries caused by inefficient delivery or reduced cellular uptake of some components. Therefore, there is a significant practical advantage for ongoing experimental research into localized molecular devices.

Designing localized molecular devices is, in some ways, a more complex task than engineering an equivalent solution-phase circuit. In particular, when molecular components are tethered to a nanostructure, their physical location is constrained and this restricts their ability to interact with other tethered components. This is in contrast to solution-phase systems, where the standard assumption is that geometry is unimportant and components may diffuse such that they encounter each other in whatever relative orientation is required for a reaction to take place. Thus, not only are there sequence design problems to solve but also geometric constraints that must be accounted for in the system design. As in all experimental pursuits, computational modeling and simulation offers a powerful method to easily prototype candidate designs and iterate on them more rapidly than would be possible in a purely experimental design cycle. Therefore, developing computational algorithms and tools for modeling and analyzing the behavior of complex tethered molecular systems is an important area of research. A key area of such computational tools is reaction enumeration, which is the process of automating the generation of a kinetic model of a system's behavior from a structural description of its components. Geometric enumeration allows designs to be represented as a concise structural description which can then be automatically compiled into the corresponding kinetic model, thus avoiding error-prone manual conversion processes. The contribution of this paper is to apply, for the first time, reaction enumeration techniques that account for geometry to tethered molecular systems, as outlined in Figure 1.

In previous work [12], we tackled the problem of enumerating reactions to generate chemical reaction networks (CRNs) for solution-phase circuits, checking the geometric plausibility of product structures by using a structure sampling algorithm as a constraint

solver. We have also previously used structure sampling to generate candidate structures that were used to estimate the rates of reactions between tethered structures [13]. Crucially, those structures were relatively simple and had only a single tether location for each structure. Extending that previous work, here we report a more advanced tool that can be applied to arbitrary, multi-tethered DNA strand displacement components attached to the surface of a DNA origami tile, which are more complex to sample. Our system automatically, and without further user input beyond the tether locations, accounts for the geometry of the molecular structures and enumerates the possible reactions between the structures that could be tethered at multiple locations.

2 Geometric Reaction Enumeration for Tethered Molecular Devices

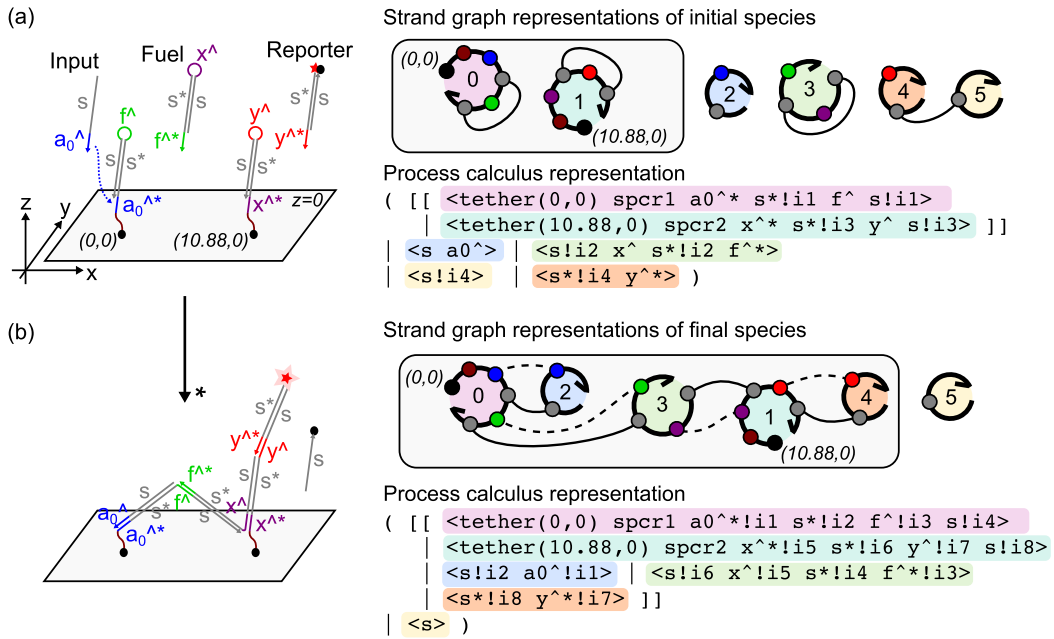
Our approach is inspired by previous work in which we used the strand graph representation of nucleic acid secondary structures [22] and defined a general-purpose framework for reaction enumeration that used a check for geometric plausibility of product structures to integrate geometric considerations directly into the reaction enumerator [12]. Here we extend that work to produce a reaction enumeration algorithm applicable to tethered molecular systems. This requires us to modify the syntax of the strand graph model, and the constraint solving procedure, to accommodate tethers. Thanks to the generality of our geometric reaction enumeration framework, the changes to the main algorithm are relatively straightforward.

2.1 Strand Graph Representations of Tethered Molecular Species

In solution phase circuits, all molecular components diffuse freely in the solution and these species can be represented as connected strand graphs [22], as all the strands in the species are connected via explicitly modeled bonds between domains. When modeling tethered circuits, however, components may be connected implicitly by both being connected to the same underlying DNA tile nanostructure, which we do not model explicitly. Therefore, we also need to include a representation of which DNA strands are attached to which tiles, and at what location, as part of the input language.

As in previous work, we use the “process calculus” representation of strand graphs as our input language. Building on the syntax introduced previously [22, 12], tethered strands are specified using angle brackets and, along with the list of domains that make up the sequence of that strand. We add “`tether(x,y)`” at either the beginning or end of the domain sequence, to mean that the strand is tethered to a tile at either the 5’ or 3’ end, respectively. Here, (x,y) are real numbers that represent the coordinates of the tether location on the underlying 2D tile surface. The coordinate system is arbitrary and we simply assume that all tethers on a single tile adhere to a common coordinate system for specifying their tether locations. Although, in practice, a single strand might have multiple tethers representing a loop out of the surface of the tile, here we assume that each strand can have only one tether.

To specify which components are bound to which tile, we must generalize the notion of a species by specifying two types: *free species* and *tile species*. Free species are those diffusing freely in the solution; they are not tethered and are simply represented as a fully-connected strand graph that cannot contain any tethers. In the process calculus syntax, species are combined using the | “vertical bar” parallel composition operator. Tile species, on the other hand, represent a two-dimensional addressable substrate, such as a DNA origami tile, which may incorporate multiple distinct components attached to the tile at different locations. Each tile species is delimited by `[` and `]` symbols and the species within are encoded using the standard process calculus syntax for single strands and the same parallel composition



■ **Figure 2** Examples of tethered strand graph representations of species before and after a partially-localized hairpin chain reaction [6] has taken place. (a) The initial state of this example system includes two hairpins tethered to a DNA origami tile at coordinates $(0, 0)$ and $(10.88, 0)$, along with freely diffusing input, fuel hairpin, and reporter complex species. The right-hand side shows strand graph representations of the individual species (connected components), with tethered species shown in the box. Individual sites are colored to match the corresponding domains in the image on the left and edges are solid for bonds between long domains and broken for edges between toehold domains. A process calculus representation of the system is shown, with pieces of syntax color-coded to match the background color of the corresponding strand in the strand graph representation. (b) Similar representation of the final state of this system; note in particular that some strands that were previously free are now tethered to the tile because they are bound to the tethered hairpins.

operator. As there can be multiple tethered species on each tile, tile species objects must therefore store these tethered species as a list of individual strand graph objects, each of which must be a fully connected individual strand graph and each of which must contain at least one tether specification. This is illustrated in Figure 2, which shows the process calculus representation, and connected strand graphs for the individual species, in a simple localized hairpin reaction [6].

2.2 A Geometric Reaction Enumeration Algorithm for Tethered Systems

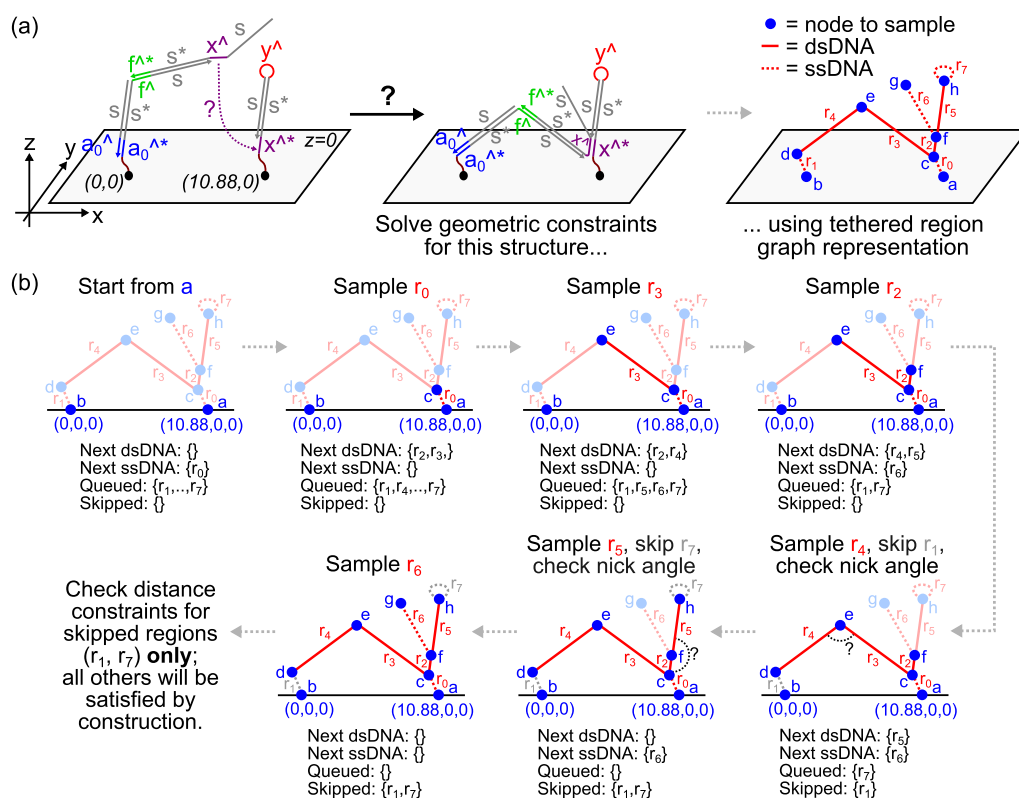
In addition to modifying the syntax of strand graphs to accommodate tether specifications, we must also modify the reaction enumeration algorithm from our previous work [12] to account for the fact that some components are attached to tiles. Experimental work has shown that components attached to different tile species can in fact interact with each other, however, this undesirable interaction can be reduced by lowering the concentrations of the tiles [6]. Here, therefore, we do not allow components attached to different tiles to interact with each other. Thus, we had to modify the pre-existing reaction enumeration algorithm with these two different classes of species (free species and tile species) and enforce the constraint that while two free species may interact and a free species may interact with a tile

$$\begin{array}{c}
\frac{S_i \in \bar{S} \quad S_i \rightarrow G' \quad \bar{S}' = \bar{S} \setminus \{S_i\} \cup \text{tethered}(G')}{[[\bar{S}]] \Rightarrow [[\bar{S}']] + \text{free}(G')} \quad (\text{TILEONE}) \\
\\
\frac{S_i, S_j \in \bar{S} \quad i \neq j \quad (S_i \parallel S_j) \rightarrow G' \quad \bar{S}' = \bar{S} \setminus \{S_i, S_j\} \cup \text{tethered}(G')}{[[\bar{S}]] \Rightarrow [[\bar{S}']] + \text{free}(G')} \quad (\text{TILETWO}) \\
\\
\frac{S_i \in \bar{S} \quad (S_i \parallel S) \rightarrow G' \quad \bar{S}' = \bar{S} \setminus \{S_i\} \cup \text{tethered}(G')}{[[\bar{S}]] + S \Rightarrow [[\bar{S}']] + \text{free}(G')} \quad (\text{TILEFREE}) \\
\\
\frac{(S \parallel S') \rightarrow G' \quad \text{tethered}(G') = \emptyset}{S + S' \Rightarrow \text{free}(G')} \quad (\text{FREEFREE})
\end{array}$$

■ **Figure 3** Inference rules defining a geometrically-aware transition relation (\Rightarrow) between collections of strand graph species, including tile species with tethered components. These rules build on our previously reported \rightarrow rules for geometrically aware reaction enumeration on strand graphs of free species [12], which are reproduced as Figure A.1 in the Appendix for convenience and which incorporate a *plausible* predicate to encode geometric plausibility. The rules presented here implement a CRN semantics [22] for systems that may contain both free and tethered species, where we write \bar{S} for finite collections of fully connected strand graphs, each of which represents a single molecular species. The syntax $[[\bar{S}]]$ represents a tile species containing species \bar{S} , each of which must be tethered; we assume that other species are freely diffusing. We write $S \parallel S'$ for the strand graph resulting from the parallel composition of S and S' . We write $\text{tethered}(G)$ and $\text{free}(G)$ for the connected components of a strand graph G that represent tethered and free species, that is, those species which do or do not include a tether specification, respectively. Rule (TILEONE) identifies a single species tethered to the same tile that may react. Similarly, rule (TILETWO) encodes a reaction between two species tethered to the same tile. Rule (TILEFREE) identifies a free species that may react with a species tethered to a tile, and rule (FREEFREE) lifts the standard rules for reactions involving two free species into the \Rightarrow relation. All reactions involving at least one tile species may produce new tethered species, and possibly some free species also. In the case of rule (FREEFREE), we note that the $\text{tethered}(G') = \emptyset$ premise is vacuously true since a tethered species cannot spontaneously emerge from a reaction between two free species. These rules form the basis of our localized geometric reaction enumerator.

species, two separate tile species cannot interact. This means there are three possible kinds of binding interaction that our reaction enumerator may identify: (i) interactions between two free species, (ii) interactions between a free species and a species tethered to a tile, and (iii) interactions between two components tethered to *the same tile*. In the latter case, the enumerator must check for binding interactions between all pairs of species attached to that tile, and each such interaction then becomes a unimolecular reaction whose reactant is the entire species due to the fast timescale of localized reactions. These may be unimolecular (e.g., toehold unbinding, strand displacement, or branch migration) or bimolecular (e.g., binding) from the perspective of the components attached to the tile. However, these are all considered as unimolecular interactions from the perspective of an observer of the whole system, due to the fast timescale of the localized reactions. The sole reactant of such reactions is the initial tile species, and such reactions will produce a new tile species and may also produce free species, e.g., if a strand unbinds or is displaced from the tile surface.

The inference rules that formally specify these interactions build on those from our previous work [12], which are reproduced in Figure A.1 in the Appendix. The new rules are presented in Figure 3 and serve as the basis for our geometric reaction enumeration algorithm.



■ **Figure 4** Region graph formation and structure sampling for tethered systems. (a) For a candidate localized toehold-binding reaction in a hairpin chain reaction system [6], the region graph is formed for the putative product structure, following the scheme from our previous work [12] with the addition of tether information. (b) Example of structure sampling across the tethered region graph from part (a). At each step, tile constraints are checked to ensure the structure does not “pass through” the tile. Angle constraints at nicks are checked when both sides of a double-stranded nick are sampled. Distance constraints need only be checked explicitly for regions that are “skipped” because a coordinate has already been generated for both ends when it is selected for processing.

This follows a standard recursive scheme [14] that checks for possible reactions between known species in a queue, adds them to the CRN, and identifies new species produced which are then added to the queue for be checked for further possible interactions. The requirement for constraints imposed by molecular geometry to be satisfied is encoded in the *plausible* predicate. The implementation of this predicate to account for constraints arising from tethering of components onto tile surfaces, a novel component of this work, is outlined below.

2.3 Geometric Constraints for Tethered Systems

Each candidate structure to be tested for geometric plausibility is converted into a set of geometric constraints that represents the possible geometry of the structure. When all constraints in this set of constraints are simultaneously satisfiable, the structure is deemed to be geometrically plausible. Building on previous work [12], these strand graphs are translated to undirected *region graphs* by condensing the domain-based representation, including information on tether coordinates, as illustrated in Figure 4(a) for a candidate localized hairpin reaction [6]. The vertices of this region graph represent the locations on

the structure for which we must determine the physical coordinates: the junctions between double-stranded and single-stranded DNA, and nicks between double-stranded regions, and strand termini. The biophysical model that we use is deliberately simple: double-stranded DNA is modeled as rigid rods and single-stranded DNA is modeled as an infinitely flexible polymer chain. As outlined below, in some tests we restrict the flexibility of nicks. Unless stated otherwise below, we assume the length per nucleotide to be 0.68 nm in a single-stranded region and 0.34 nm in a double-stranded region, following previous standard practice from the literature [10, 7]. Under this biophysical model, we must check satisfaction of several kinds of geometric constraint derived from the tethered region graph:

- *Domain length constraints:* as in our previous work [12], the key constraints arise from the lengths of domains. As outlined below, constraints arising from hybridization between complementary domains will be dealt with implicitly, as will many domain length constraints. For rigid double-stranded domains, the length will be fixed; for flexible single-stranded domains, it must fall between zero and the maximum length.
- *Tether and tile constraints:* A tether anchors one end of a domain at a fixed location on the tile surface. We assume that the tile surface lies in the $z = 0$ plane in a coordinate system defined by the tile itself, and all components are tethered on the same side of the tile surface and protrude in the positive z side. Tethers on a strand are constrained to specific x and y coordinates, with $z = 0$. Non-tethered positions on a strand are constrained to have a non-negative z coordinate, which ensures that no part of the tethered structure may “pass through” the tile surface.
- *Angle constraints at nicks:* Nicks between double-stranded regions give the structure some degree of flexibility, but not as much as for a fully single-stranded region. As discussed below, this can lead to some reactions being identified that one might not expect and which are arguably unphysical, as they require extreme flexibility at the nick. In some tests, therefore, we limit the range of angles through which a nick may bend, given previous work on the flexibility of nicks [6], by allowing the maximum angle permitted between nicked domains to be tuned. The angles are the angle of deviation ϕ between the regions where they meet at the nick: this is computed from the directionality vectors v_1 and v_2 of the two regions, as $\phi = \cos^{-1}((v_1 \cdot v_2)/(|v_1||v_2|))$.

2.4 Structure Sampling for Tethered Geometric Constraints

After generating constraints, we attempt to find a possible candidate solution using a randomized structure sampling approach built upon that described in our previous work [12], in which we randomly generate structures that satisfy many of the constraints by construction, and then check whether the remaining constraints are satisfied. Figure 4(b) presents an example structure sampling run over a tethered region graph. This illustrates the major change to the structure sampling algorithm developed in this work, which is the choice of where to start sampling the structures of components attached to a tile species. We generalized the pre-existing algorithm by defining a “starting vertex set”, which in the non-tethered case is just one of the vertices with a maximal degree, chosen at random and assigned the origin coordinates $(0, 0, 0)$. In the tethered case, however, we assign the starting vertex set to be the set of tethered vertices, each of which is assigned the corresponding tether coordinate, and choose one at random to be the starting vertex. The rationale for using the tethers as the starting points is that their coordinates are fixed, so it is preferable to start sampling from these coordinates and move away from them rather than fixing some other point as the origin, as it is highly unlikely that the sampled structure will end up being close enough to one of the tethers for the tether constraints to be satisfied. Intuitively, the tile acts as an implicit loop which imposes geometric constraints on the attached components.

We proceed by dividing the regions into unprocessed sets of double-stranded and single-stranded regions and begin by sampling a directionality vector and length for a region chosen at random, following the heuristic that single-stranded regions are only processed when there are no double-stranded regions remaining to be processed. If a coordinate has already been sampled for the node at the far end of a candidate region then it is “skipped”: these are the regions for which the constraints must be checked explicitly at the end of the sampling process, when all nodes from the region graph have a sample coordinate. If a sampled double-stranded region forms a nick, the angle of deviation at the nick is checked against any upper limit that has been set. Lengths were fixed for double-stranded regions and chosen from a worm-like chain distribution between zero and the maximum possible length for single-stranded regions. Angles of deviation were sampled from uniform distributions over 3D polar coordinate angles, adjusted as needed for the spherical correction, with the exception of angles at nicks which were sampled from an empirical distribution of nicked angles derived from oxDNA simulations carried out by Chatterjee et al. [6]. If a maximum value for angles at nicks has been set, we check the nicked angles and reject the structure if any nicked angle constraints are not satisfied. The permissible angle of deviation at nicks can be tuned to enable a greater fraction of structures to be accepted, at the risk of permitting some unrealistic structures to be allowed. If the sampled structure satisfies all the constraints, we have found a geometrically plausible structure. If not, the randomized algorithm tries again until a predetermined number of trials is reached (typically 1000), at which point we infer that the structure is likely to be geometrically implausible. In the latter case, and thus our reaction enumerator does not add the reaction to the network.

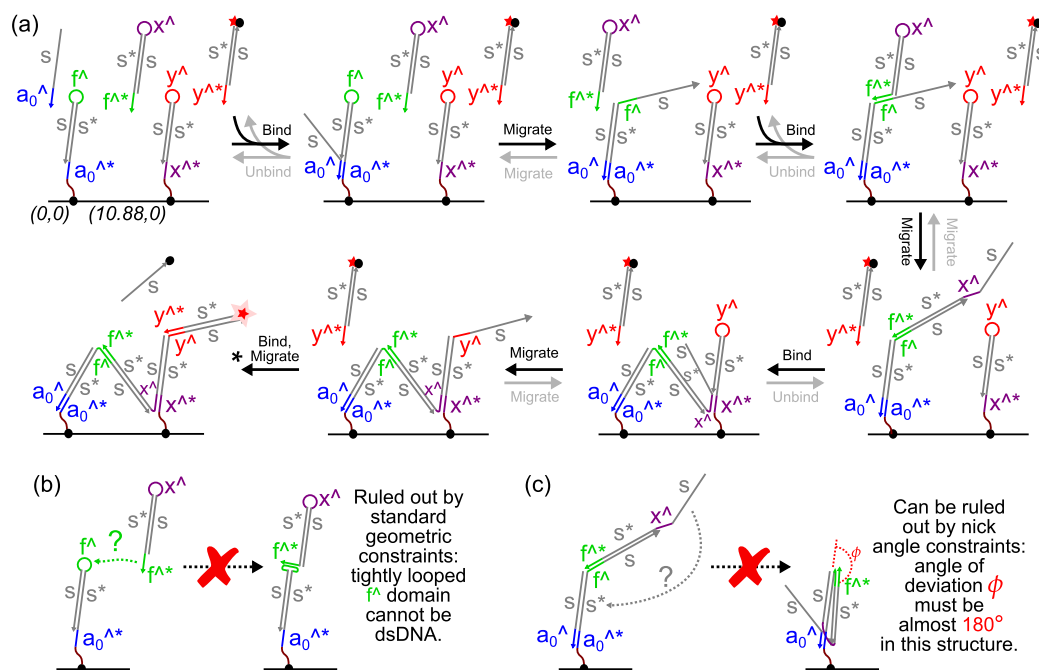
3 Results

We implemented a prototype of the algorithm presented in Section 2 above using the Python programming language and demonstrated its capabilities on several example systems from the literature. Code for our prototype system is available online under an open source license from <https://github.com/matthewlakin/LocalizedEnumerator>.

3.1 Localized Hairpin Chain Reaction Example

We first used our system to generate a kinetic model of the cascade of localized hairpin chain reactions that was experimentally realized by Chatterjee et al. [6]. In that work, where they used cascades of strand displacement reactions to build signal transmission lines and studied the effect of the spatial organization of components on circuit performance. Some of the hairpins were attached to a DNA origami surface, while others were freely diffusing in solution and served as “fuels”. We verified that our system could successfully enumerate the viable reactions in this system, based on considerations of molecular geometry.

We tested our tethered implementation on the corresponding set of localized reactions, in which every other hairpin is tethered to the surface, while the intervening “fuel” hairpins are freely diffusing in solution. We used 10.88 nm as the distance between tethered hairpins and found that our system could enumerate the same set of reactions as was experimentally observed by Chatterjee et al., as illustrated in Figure 5(a). The output for these species and reactions as generated by our prototype tool can be found in Figure A.2 in the Appendix. In a preliminary quantitative analysis, the default conditions used in our tests (1000 sampling attempts, nicked angle constraint of 105°) produced this set of reactions around 17% of the time, and decreasing the maximum nicked angle to 90° saw this increase to around 29%. (Quantitative aspects of algorithm performance are discussed further below.) When

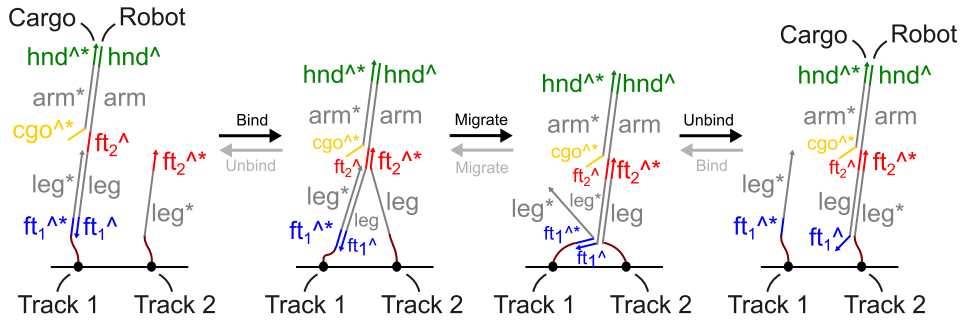


■ **Figure 5** Results from using our system to model the localized strand displacement cascade of Chatterjee et al. [6]. (a) Enumerated CRN reactions for a two-hairpin system, visualized via a redrawing of the state space produced with a single copy of each species present. (b) Example of an interaction between a closed hairpin and the fuel hairpin that is disallowed due to geometric constraints on the looped toehold in the tethered hairpin. (c) Example of a reaction that can be enumerated when nick angle constraints are not enforced: the opened hairpin folds back on itself in an arguably unphysical manner. Such reactions can be ruled out via an appropriate upper limit on the angle of deviation ϕ that is permitted at nicks.

we increased the inter-tether distance to 21.76 nm we found that, at this larger spacing, our geometric enumerator predicted that the tethered species would not interact after the fuel had bound to the first hairpin. Encouragingly, this result was consistent across all quantitative tests. Experimental work using the 21.76 nm inter-tether distance, found a significantly lower output signal, indicating little or no reactivity [6], which agreed with the results from our model. Crucially, these reactions were enumerated without any user-required specification of the possible interactions, beyond the tether coordinates provided as part of the structural design. Previous modeling work [15, 6] required the user to manually specify which tethered hairpins may interact, which introduced non-declarative, non-structural elements into the model specification.

In addition to enumerating the hairpin-opening reactions, our simplified geometric ruleset avoided leaks that might be caused by binding of a complementary toehold with the looped toehold sequestered in a closed hairpin. Importantly, this was achieved solely based on the violation of geometric constraints introduced by the looping of the toehold, rather than via rule side-conditions specially introduced to prevent binding into a loop, as in earlier work on strand graph reaction semantics [22]. An example of this kind of reaction is illustrated in Figure 5(b); this would correspond to the binding of the toehold domain $f^{\wedge*}$ from the fuel hairpin binding to the complementary looped toehold f^\wedge in the unopened localized hairpin.

As indicated above, in addition to the designed reaction pathway, our system also generated some unusual intermediate structures that led to a chain of side reactions. These intermediate structures were formed because the angle between some of the nicked structures



■ **Figure 6** Outline of key walking reaction from the cargo-sorting robot system of Thubagere et al. [28]. The robot consists of a single strand with two foot domains (ft_1^\wedge and ft_2^\wedge) connected by a leg domain. Track locations consist of complementary foot domains (alternating $ft_1^{\wedge*}$ and $ft_2^{\wedge*}$, as outlined in the main text) and a complementary leg domain. The robot moves between track locations via fully reversible toehold-mediated strand displacement reactions, as illustrated above. Here, the robot is presented with additional arm and hand (hnd^\wedge) domains for cargo handling, and is shown transporting a cargo strand that is hybridized via these domains. Cargo pickup and drop-off occurs via similar strand displacement reactions involving these domains, not shown here [28].

made them flexible enough to bend back through almost 180° and undergo strand displacement reactions with themselves, such as the example shown in Figure 5(c). While permitted by our simplified biophysical model, such reactions are highly unlikely in reality, given that nicks are not infinitely flexible, and are arguably unphysical, given steric effects. Preliminary quantitative analysis suggests that the prevalence of such reactions depends on the angle constraints imposed by the biophysical model as well as the number of sampling attempts permitted. A more permissive biophysical model, with more attempts permitted to find a solution, enables more structures and reactions to be enumerated. Thus, we were able to automatically generate appropriate models with minimal user input beyond the initial structural specification, although more work on the constraint solving algorithm may improve the reliability of the outputs, as discussed below.

3.2 Cargo Sorting Robot Example

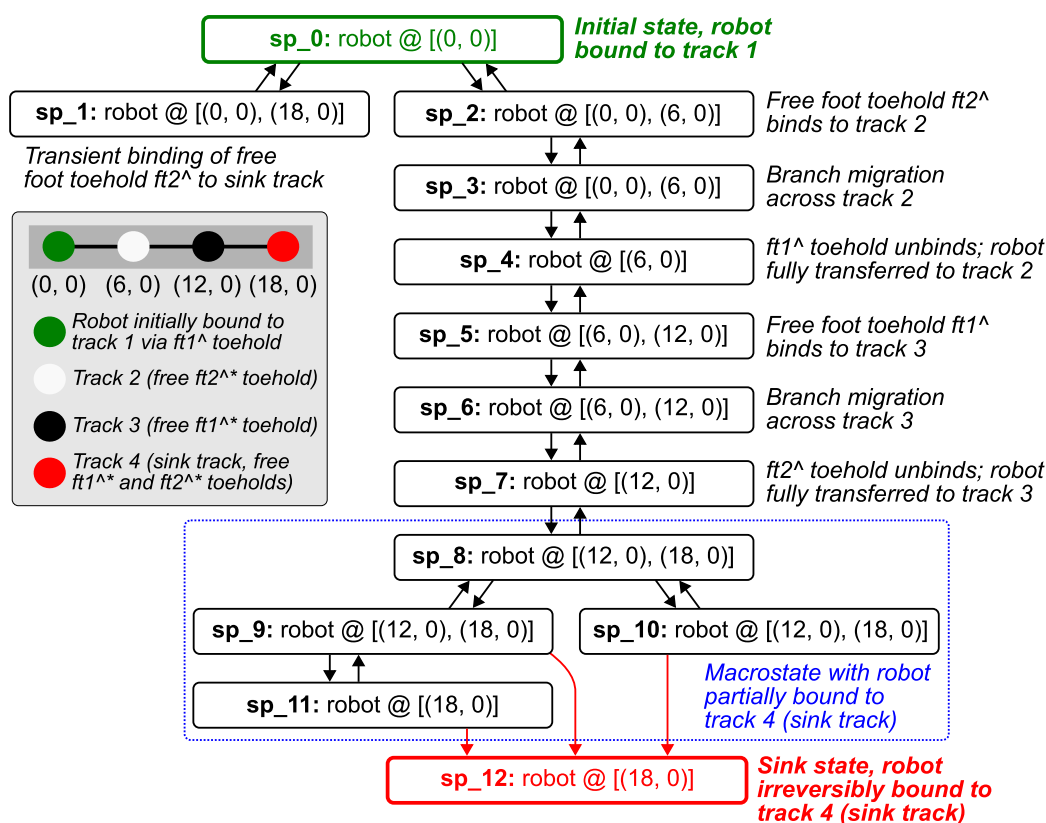
Transporting cargo against a diffusion gradient is common in biological systems. DNA-based molecular robots that can perform such tasks have been built using a variety of reaction frameworks, including enzyme-driven reactions [3], deoxyribozymes [30, 19], and DNA strand displacement reactions. Notably, Thubagere et al. [28] designed and built a simple molecular robot that performs a random walk on a two-dimensional DNA origami tile. If the robot strand encounters a *cargo strand*, the robot may pick up the cargo before continuing the random walk. If the robot encounters a *goal location*, the cargo may be dropped off at that location. The locomotion, cargo pick-up, and cargo drop-off processes are all carried out by toehold-mediated strand displacement reactions on a surface, making them ideally suited to analysis using our geometric reaction enumerator. We used our system to analyze the effects of geometric constraints imposed by the distances between track locations and domain lengths on the robot's random walk as well as the picking-up and dropping-off of any cargo. We show that our system can automatically determine the possible paths a robot can take by essentially enumerating the possible state space of surface-bound reactions; we note that simulations were carried out by Thubagere et al. [28] using a manually derived model.

Figure 6 illustrates the robot, which is a single strand of DNA with two toeholds serving as *foot* domains and one long domain serving as the *leg*. We modeled robots with two distinct foot domains (\hat{ft}_1 and \hat{ft}_2), to reduce the frequency of track-skipping behavior in which the robot could interact with multiple nearby tracks, not just the closest one. Track-skipping significantly increases the number of possible states due to transient toehold-binding interactions, making the enumeration slower and the results more difficult to interpret. Figure 6 also illustrates the reactions that implement the robot's random walk. To step from one track to another, the free foot domain of the robot first binds to the complementary foot domain of another track strand. The complementary leg domains of the two track strands then compete to bind to the leg domain of the robot in a branch migration reaction. When this process fully displaces the robot from the previous track strand, the previously bound foot domain can dissociate, completing the translocation of the robot from the first track location to the second. The ability of the robot to interact with different track locations at once is geometrically constrained, thus making it an ideal test of our geometric constraint solver. In addition, because the robot is designed to be bound to at most two tethers at a time, so structure sampling is a very tractable way to carry out the analysis. We studied both simple linear and diagonally offset tracks, with a 6 nm distance between track locations.

In contrast with our biophysical model outlined above, Thubagere et al. [28] assumed the length per nucleotide of single-stranded DNA to be 0.43 nm. To demonstrate the geometric analysis capabilities of our system, we used it to enumerate reactions for the molecular robot system using both biophysical models. Using a track containing five locations 6 nm apart, we found more possible states for a length of 0.68 nm per nucleotide than for 0.43 nm. This is not surprising as, when the length per nucleotide is greater, domains of a given length are longer with respect to the fixed length between track locations, so the strands can reach further to find and interact with a complementary domain. Figure A.3 in the Appendix compares CRNs generated for a random walk on a track with five sites for both the nucleotide lengths. For consistency with Thubagere et al.'s work [28], we assumed a length of 0.43 nm per single-stranded nucleotide in the rest of our reaction enumeration experiments on the molecular robot system.

We then used our geometric reaction enumerator to analyze the possible reactions arising from the robot's random walk in more detail. Figure 7 presents aCRN enumerated for a robot walking across two tracks to a sink state, as shown in the inset of that Figure. The sink state includes an additional complementary toehold that binds to both foot domains of the robot and sequesters it permanently attached to the sink state track, meaning there will be no further reactions. Since all components in this system are tethered to the same surface, all reactions will be unimolecular reactions involving tile species and there will be a single copy of a single tile species present at any one time. Thus, the resulting CRN in this case coincides precisely with the state space of the system. Each species in Figure 7 is annotated with location of the robot in each state, determined via the coordinates of the tethers to which it is connected.

Initially, the robot is attached to the track at location $(0, 0)$, corresponding to the tile species sp_0 . Then, the free foot domain of the robot binds to the complementary foot domain of one of the tracks: either one of the other tracks with a complementary toehold, either at $(6, 0)$ (the neighboring track where we intend the robot to step next), or $(18, 0)$ (the sink state). Note that the $(12, 0)$ track is not an option because it binds to the other foot domain. Any interaction with the sink state track is only transient, however, because while that track is close enough for the robot's foot domain to bind to the sink state (sp_1), the tracks are not close enough for the subsequent strand displacement reaction that would be



■ **Figure 7** CRN resulting from enumeration of molecular robot reactions walking across a series of four track locations arranged linearly (track layout illustrated in inset). Each species corresponds to a state of the tile species, which are annotated with the tether locations to which the robot strand is connected. Red colored arrows represent the irreversible reaction when the robot completes its final step into the sink state. Species are annotated to describe the reactions occurring at each step; note in particular the multiple pathways into the sink state, as this translocation required branch migration across several domains.

required for the robot to transfer completely to the sink state track. Thus, the reaction that moves from sp_0 to sp_1 is reversible, and no further reactions possible in that branch of the CRN. Such transient toehold binding interactions are likely undesirable in the kinetic model, and it is interesting that our model enumerates a number of these. This suggests that, if our model is accurate, the walking behavior of these robots may be somewhat complex. In principle, our prototype reaction enumerator could be modified to remove such “unproductive” toehold binding reactions, as has been done in previous work on reaction enumeration in simpler strand displacement systems [17].

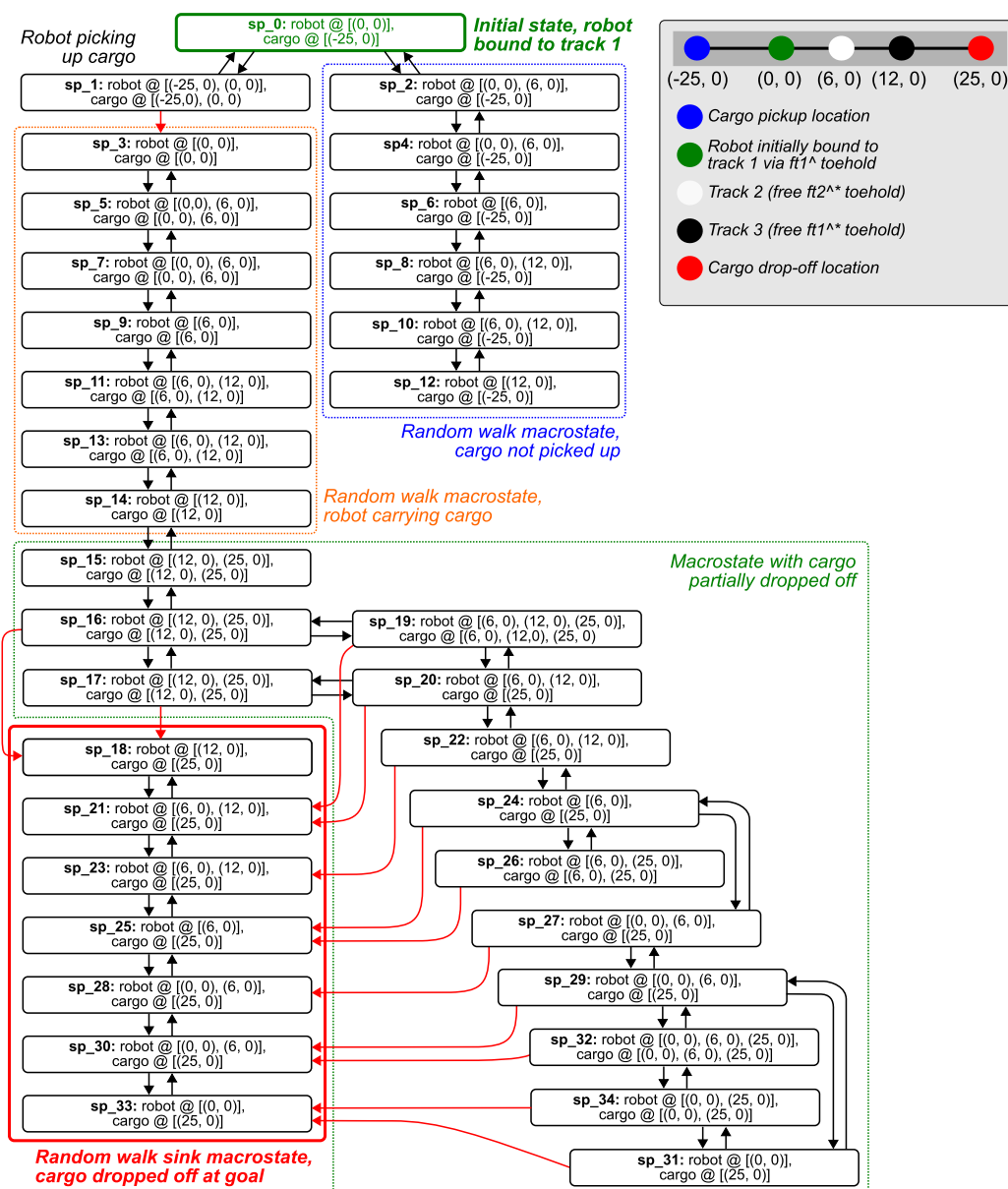
The other possible reaction for species sp_0 is for the unbound foot domain to bind to the adjacent track location at $(6, 0)$; this initiates the desired pathway of reactions that leads to the robot to be fully transferred to this location (sp_4). Having moved to the second track location, species sp_5 through sp_7 are a linear sequence corresponding to the robot moving to the third location. All of these reactions are reversible, meaning that we are really seeing a random walk, as expected. After species sp_8 , however, the states branch as the robot is now close enough that it can interact with the sink state track location. Because our compiler does not condense fast reactions into a single step, as has been done in previous

work [1, 17], there are multiple possible strand displacement pathways that the system can follow to reach the final state, species `sp_12`, where the robot is fully and irrevocably bound to the sink state track. This is because there are two domains that must be displaced and each of these is enumerated as a separate strand displacement reaction in our system. While simple, this example demonstrates that our geometric reaction enumeration system can be used to analyze the behavior of surface-bound molecular robotics systems that function using DNA strand displacement reactions.

We also used our system for the robot's random walk on a more complex track, which consists of two sets of track locations, spaced 6 nm apart and offset diagonally into a hexagonal lattice pattern. This increased the total number of track locations, which, as expected, led to an increase in the number of enumerated reactions, including transient side-reactions. This state transition diagram is presented in Figure A.4 in the Appendix. Removing unproductive toehold binding reactions should help to reduce the number of possible side-reactions but, in general, this example serves to illustrate the complex nature of localized interactions and the need for careful circuit design to manage this complexity, which might not be apparent from experimental observations.

To model cargo pickup and drop-off reactions, we designed a simplified track (Figure 8, inset) to minimize the number of possible cross-reactions and thus simplify the analysis. The robot and cargo components were as illustrated in Figure 6. The robot's initial position is at the origin and the cargo was placed at a distance far enough away that the robot can only interact with the cargo strand when the robot is at the initial position. The cargo drop-off (goal) location was placed similarly at the far end of a short track, at (25, 0), such that the cargo drop-off reactions can only occur when the robot is at the far end of the track. (In fact, tests showed that the robot can drop off cargo at a goal positioned as far away as (32.8, 0).)

Figure 8 presents a CRN enumerated for this system, annotated with the robot and cargo positions in each species and with a number of distinct macro-states identified. These macro-states can clearly be identified within the resulting CRN. Species `sp_0` is the initial state, and in the top right of Figure 8 we observe a reversible set of states corresponding to an initial random walk of the robot back and forth along the track, without having first picked up the cargo. The left-hand branch via species `sp_1` represents the robot picking up the cargo, which is completed via the irreversible reaction from species `sp_1` to species `sp_2`: once the cargo has been picked up from its initial position, it cannot be put back down except at the goal location. There is then a similar random walk along the track, but this time with the cargo attached to the robot. The blue-outlined macrostate corresponds to the robot attempting to drop off the cargo at the goal. The cargo drop-off reaction is irreversible, requiring strand displacement across two domains sequentially; it thus causes branching in the state graph, as in our model the robot may interact with other tracks while the drop-off reaction is only partially completed. The branching is made worse in this case by the fact that the longer robot and cargo may also undergo unproductive, transient toehold-binding interactions with other strands in the system, similar to those discussed above. However, there is a cut through the graph of irreversible reactions leading to the red-outlined sink macrostate, in which the cargo has been irreversibly dropped off at the goal and the robot goes back to a random walk between the tracks, without the cargo which is now attached to the goal. This CRN thus recapitulates the expected behavior of the cargo-sorting molecular robot system and demonstrates that our system is capable of automated analysis of such surface-bound nanoscale molecular robotics systems.



■ **Figure 8** CRN resulting from enumeration of the molecular robot picking up cargo, walking on a track, and dropping the cargo off at a goal location (track layout illustrated in inset). Each species corresponds to a state of the tile species, which are annotated with the tether locations to which the robot and cargo strands are connected. Red colored arrows represent irreversible reactions; species and macrostates are annotated as appropriate to illustrate the behaviors of the robot.

4 Related Work

The work outlined here builds on our own previous work on geometrically-aware enumeration of DNA strand displacement reactions [12] and localized DNA reaction systems [13, 16]. Some of Lakin’s early work in this area [15] was integrated into the Visual DSD reaction enumerator system [18, 17, 27] and was used to model the localized hairpin reaction system

experimentally demonstrated by Chatterjee et al. [6], which we also study here. The key difference between that work and the system reported here is that the earlier versions required the user to manually specify, via a system of “tags”, which tethers could interact with each other and at what rate. The advantage of the approach outlined here is that no information beyond the structural model of the system is required to enumerate reactions while accounting for geometric constraints.

Other related work on computational approaches to reaction enumeration in DNA strand displacement systems includes the Peppercorn reaction enumerator [1] and Nuskell compiler pipeline [2], both developed by the Winfree group. In addition, our work is inspired by previous approaches to automated analysis of localized molecular circuits and walkers [7, 8], and by earlier work on physics-based modeling of molecular robotics systems [24]. Other approaches to model such systems have included coarse-grained models [20] such as oxDNA [9]. Finally, other related work on localized systems beyond the systems explicitly modeled in this paper [6, 28] have included similar work on localized DNA hairpin circuits [4, 5] as well as other kinds of molecular robots [11, 32, 19]; the latter could be modeled using enumerator extensions that incorporate non-DNA-based chemical reactions into DSD models, similar to those previously developed for the Visual DSD system [33].

5 Discussion

In conclusion, we modeled DNA nanodevices tethered to tiles by implementing a localized variant of our geometric reaction enumerator for solution-phase circuits [12]. We developed new syntax and semantics for tethered components and species representing DNA tiles in our implementation and modified our structure sampling approach to sample starting from the tether coordinates in an attempt to find a geometrically plausible structure. This tool that can be used to model systems *in silico* to rapidly rule out designs that may lead to undesirable reactions, based in particular on their molecular geometry.

We demonstrated this system by first modeling the hairpin chain reaction signal transmission line reported by Chatterjee et al [6] and found results consistent with their experimental work, with hairpins able to interact when positioned at a 10.88 nm distance but not at 21.76 nm. Interestingly, our model identified some apparently unlikely reactions that seemingly arise due to excessive flexibility at nicks between double-stranded regions, which allow the structures to bend far enough to interact with themselves and thus trigger unexpected interactions. Our preliminary quantitative analysis found that the prevalence of such reactions is dependent on the permissiveness of the biophysical model with regard to flexibility at nicks, which permits a greater range of structures, some of which may be unphysical in reality. It also depends on the number of sampling attempts permitted, as allowing more attempts make it more likely that such structures will be found during any given sampling run. However, at the 21.76 nm spacing, no such reactions were enumerated. This indicates that our algorithm appears to be sound in the sense that it does not identify any impossible reactions. Nevertheless, work remains to understand the behavioral tradeoffs in the system and to tune the algorithm to maximize its reliability when faced with such structures that are technically feasible but unlikely to be sampled and arguably unphysical. Potential new approaches to address this issue include attempting to convert unsuccessful sampling attempts into satisfying structures, as outlined below.

We also demonstrated that our system could model more sophisticated systems like the cargo-sorting robot system of Thubagere et al. [28]. Our analysis showed that enumerating even a small number of tracks could lead to an explosion of reactions, though these could

in principle be filtered out, for example, by merging fast reactions to eliminate short-lived intermediate species from the model altogether based on the separation of timescales. This has been done in previous work [1, 17] and was not done here only because re-implementing such algorithms is not the core of the work reported here. Our analysis also identified the possibility of track-jumping behavior, whereby robots may interact with components that are not directly adjacent to their current track location. Because of our alternating toehold domains on the tracks and robot, these interactions are typically just transient and “unproductive” toehold-binding reactions. Nevertheless, these species exist in the model and thus contribute to a combinatorial explosion in the number of possible states as the number of tracks increases, which complicates interpretation of the resulting CRN. In addition, we could automatically filter out some of these toehold-only interactions as in previous work [17], although it may not be as straightforward to determine which reactions are unproductive in the more general strand graph-based model than in simpler systems [17].

We deliberately simplified our biophysical model so that structures can be converted to constraint problems that can be solved using relatively straightforward algorithms. Inspired by our previous work on geometric reaction enumeration [12], we used a structure sampling approach to solve the constraints [13]. Importantly, as the complexity of the structure increases, more sampling trials may be required to find a satisfying set of coordinates, because each pair of tethers creates a loop in the structure (with the tile surface implicitly completing the loop) that imposes constraints that must all be satisfied. This was reflected in our preliminary quantitative analysis, in which some unlikely structures would be found in some attempts but not in others. In particular, finding a valid sampling that satisfies the whole structure becomes less likely as the number of tethers increases. In the initial work presented here, we fixed a seed for reproducibility of results; our preliminary quantitative analysis used a larger set of seeds to produce some initial statistics on system behavior. Building on this work, an important future direction will be to optimize the system for efficient constraint-solving on larger structures containing multiple tethers, to increase the probability of finding a solution if one exists and to increase the reliability of the system across different sampling attempts. For example, we could prune parts of the graph that do not impact plausibility, such as those that do not contain loops. Alternatively, we could decompose the graph into smaller subgraph, try to sample them separately, and then attempt to compose the solutions to assemble a satisfying set of coordinates for the whole graph. Finally, as mentioned above, rather than simply abandoning unsuccessful sampling attempts, we could instead use them as a starting point for other optimization algorithms, such as the distance matrix completion algorithm [31], to try to improve them to produce a satisfying set of coordinates for the structure. This approach is promising because our structure sampling algorithm ensures that the majority of the constraints in the system are already satisfied by construction, and should reduce the variability in the results obtained from our system.

A key issue not addressed by the work presented here is the estimation of rate constants for localized reactions. This is a challenging problem; experimental work enabled the estimation of localized rate constants for the hairpin system studied here [6], using the local concentration approach to relate localized reaction kinetics to those measured in bulk solution [10, 7]. Thubagere et al. [28] also estimated rate constants for their molecular robot system. We previously used the local concentration approach to automate the estimation of the local concentration for a localized toehold binding reaction [13], and anticipate that this, or a similar approach, could be integrated with our reaction enumerator in future work to automate the generation of realistic kinetic models for localized systems, perhaps based on a statistical analysis of the proportion of successful sampling attempts.

We have assumed that the underlying nanostructure is a two-dimensional tile surface. This could be generalized to model a three-dimensional nanostructure, such as a wireframe shape [36], however, care would need to be taken to ensure that additional steric constraints imposed by the three-dimensional nanostructure do not prevent the enumerated reactions occurring. This might be achieved by explicitly modeling the underlying nanostructure and solving additional constraints to verify that a trajectory of plausible structures exists to connect the initial and final states involved in a reaction. In addition, our model does not allow components from two different tiles to interact with each other; this might be modified, as experimental work has shown that components attached to tiles can cross-react [6], while others have demonstrated programmable assembly and disassembly of DNA nanostructures [34, 21]. The latter typically involve reactions attached onto the *edges* of the tile, which might be accommodated in a more general version of the system presented here. This would require a syntax for components attached to the edge of a tile and a semantics for reactions in which two tiles are combined into one or a single tile is broken apart.

Finally, an important development of this work would be to use a more detailed biophysical model as the basis of our structural analysis. The accuracy of the model could be enhanced by considering the physical size of the DNA duplex, the twist of the two strands, and more accurate physical parameters based on DNA structural studies. Constraints on such structures might be solved by integrating more detailed structural models such as the oxDNA coarse-grained molecular dynamics system [25, 9], which could be used to parameterize more abstract domain-level kinetic models. Thus, the work reported here could serve as the basis for future multiscale modeling of DNA-based molecular systems.

References

- 1 Stefan Badelt, Casey Grun, Karthik V. Sarma, Brian Wolfe, Seung Woo Shin, and Erik Winfree. A domain-level DNA strand displacement reaction enumerator allowing arbitrary non-pseudoknotted secondary structures. *Journal of the Royal Society Interface*, 17(167):20190866, 2020. doi:10.1098/rsif.2019.0866.
- 2 Stefan Badelt, Seung Woo Shin, Robert F. Johnson, Qing Dong, Chris Thachuk, and Erik Winfree. A general-purpose CRN-to-DSD compiler with formal verification, optimization, and simulation capabilities. In R. Brijder and L. Qian, editors, *Proceedings of the 23rd International Conference on DNA Computing and Molecular Programming*, volume 10467 of *Lecture Notes in Computer Science*, pages 232–248, 2017. doi:10.1007/978-3-319-66799-7_15.
- 3 Jonathan Bath, Simon J. Green, Katherine E. Allen, and Andrew J. Turberfield. Mechanism for a directional, processive, and reversible DNA motor. *Small*, 5(13):1513–1516, 2009. doi:10.1002/smll.200900078.
- 4 Hieu Bui, Vincent Miao, Sudhanshu Garg, Reem Mokhtar, Tianqi Song, and John Reif. Design and analysis of localized DNA hybridization chain reactions. *Small*, 13(12):1602983, 2017. doi:10.1002/smll.201602983.
- 5 Hieu Bui, Shalin Shah, Reem Mokhtar, Tianqi Song, Sudhanshu Garg, and John Reif. Localized DNA hybridization chain reactions on DNA origami. *ACS Nano*, 12(2):1146–1155, 2018. doi:10.1021/acsnano.7b06699.
- 6 Gourab Chatterjee, Neil Dalchau, Richard A. Muscat, Andrew Phillips, and Georg Seelig. A spatially localized architecture for fast and modular DNA computing. *Nature Nanotechnology*, 12:920–927, 2017. doi:10.1038/nnano.2017.127.
- 7 Neil Dalchau, Harish Chandran, Nikhil Gopalkrishnan, Andrew Phillips, and John Reif. Probabilistic analysis of localized DNA hybridization circuits. *ACS Synthetic Biology*, 4(8):898–913, 2015. doi:10.1021/acssynbio.5b00044.

- 8 Frits Dannenberg, Marta Kwiatkowska, Chris Thachuk, and Andrew J. Turberfield. DNA walker circuits: computational potential, design, and verification. *Natural Computing*, 14:195–211, 2015. doi:10.1007/s11047-014-9426-9.
- 9 Jonathan P. K. Doye, Thomas E. Ouldridge, Ard A. Louis, Flavio Romano, Petr Šulc, Christian Matek, Benedict E. K. Snodin, Lorenzo Rovigatti, John S. Schreck, Ryan M. Harrison, and William P. J. Smith. Coarse-graining DNA for simulations of DNA nanotechnology. *Physical Chemistry Chemical Physics*, 15:20395–20414, 2013. doi:10.1039/C3CP53545B.
- 10 Anthony J. Genot, David Yu Zhang, Jonathan Bath, and Andrew J. Turberfield. Remote toehold: A mechanism for flexible control of DNA hybridization kinetics. *Journal of the American Chemical Society*, 133(7):2177–2182, 2011. doi:10.1021/ja1073239.
- 11 Cheulhee Jung, Peter B. Allen, and Andrew D. Ellington. A simple, cleated DNA walker that hangs on to surfaces. *ACS Nano*, 11(8):8047–8054, 2017. doi:10.1021/acsnano.7b02693.
- 12 Sarika Kumar and Matthew R. Lakin. A geometric framework for reaction enumeration in computational nucleic acid devices. *Journal of the Royal Society Interface*, 20(208):20230259, 2023. doi:10.1098/rsif.2023.0259.
- 13 Sarika Kumar, Julian M. Weisburd, and Matthew R. Lakin. Structure sampling for computational estimation of localized DNA interaction rates. *Scientific Reports*, 11:12730, 2021. doi:10.1038/s41598-021-92145-8.
- 14 Matthew R. Lakin, Loïc Paulevé, and Andrew Phillips. Stochastic simulation of multiple process calculi for biology. *Theoretical Computer Science*, 431:181–206, 2012. doi:10.1016/j.tcs.2011.12.057.
- 15 Matthew R. Lakin, Rasmus Petersen, Kathryn E. Gray, and Andrew Phillips. Abstract modeling of tethered DNA circuits. In Satoshi Murata and Satoshi Kobayashi, editors, *Proceedings of the 20th International Conference on DNA Computing and Molecular Programming*, volume 8727 of *Lecture Notes in Computer Science*, pages 132–147. Springer International Publishing, 2014. doi:10.1007/978-3-319-11295-4_9.
- 16 Matthew R. Lakin and Andrew Phillips. Automated analysis of tethered DNA nanostructures using constraint solving. *Natural Computing*, 17(4):709–722, 2018. doi:10.1007/s11047-018-9693-y.
- 17 Matthew R. Lakin, Simon Youssef, Luca Cardelli, and Andrew Phillips. Abstractions for DNA circuit design. *Journal of the Royal Society Interface*, 9(68):460–486, 2012. doi:10.1098/rsif.2011.0343.
- 18 Matthew R. Lakin, Simon Youssef, Filippo Polo, Stephen Emmott, and Andrew Phillips. Visual DSD: a design and analysis tool for DNA strand displacement systems. *Bioinformatics*, 27(22):3211–3213, 2011. doi:10.1093/bioinformatics/btr543.
- 19 Kyle Lund, Anthony J. Manzo, Nadine Dabby, Nicole Michelotti, Alexander Johnson-Buck, Jeanette Nangreave, Steven Taylor, Renjun Pei, Milan N. Stojanovic, Nils G. Walter, Erik Winfree, and Hao Yan. Molecular robots guided by prescriptive landscapes. *Nature*, 465:206–210, 2010. doi:10.1038/nature09012.
- 20 Thomas E. Ouldridge, Rollo L. Hoare, Ard A. Louis, Jonathan P. K. Doye, Jonathan Bath, and Andrew J. Turberfield. Optimizing DNA nanotechnology through coarse-grained modeling: A two-footed DNA walker. *ACS Nano*, 7:2479–2490, 2013. doi:10.1021/nn3058483.
- 21 Jennifer E. Padilla, Ruojie Sha, Martin Kristiansen, Junghuei Chen, Natasha Jonoska, and Nadrian C. Seeman. A signal-passing DNA-strand-exchange mechanism for active self-assembly of DNA nanostructures. *Angewandte Chemie International Edition*, 54(20):5939–5942, 2015. doi:10.1002/anie.201500252.
- 22 Rasmus L. Petersen, Matthew R. Lakin, and Andrew Phillips. A strand graph semantics for DNA-based computation. *Theoretical Computer Science*, 632:43–73, 2016. doi:10.1016/j.tcs.2015.07.041.
- 23 Paul W. K. Rothemund. Folding DNA to create nanoscale shapes and patterns. *Nature*, 440:297–302, 2006. doi:10.1038/nature04586.

- 24 Sudheer Sahu, Bei Wang, and John H. Reif. A framework for modeling DNA based molecular systems. *Journal of Computational and Theoretical Nanoscience*, 5(11):2124–2134, 2008. doi:10.1166/jctn.2008.1108.
- 25 Aditya Sengar, Thomas E. Ouldridge, Oliver Henrich, Lorenzo Rovigatti, and Petr Šulc. A primer on the oxDNA model of DNA: When to use it, how to simulate it and how to interpret the results. *Frontiers in Molecular Biosciences*, 8:693710, 2021. doi:10.3389/fmolb.2021.693710.
- 26 Friedrich C. Simmel, Bernard Yurke, and Hari R. Singh. Principles and applications of nucleic acid strand displacement reactions. *Chemical Reviews*, 119(10):6326–6369, 2019. doi:10.1021/acs.chemrev.8b00580.
- 27 Carlo Spaccasassi, Matthew R. Lakin, and Andrew Phillips. A logic programming language for computational nucleic acid devices. *ACS Synthetic Biology*, 8(7):1530–1547, 2019. doi:10.1021/acssynbio.8b00229.
- 28 Anupama J. Thubagere, Wei Li, Robert F. Johnson, Zibo Chen, Shayan Doroudi, Yae Lim Lee, Gregory Izatt, Sarah Wittman, Niranjana Srinivas, Damien Woods, Erik Winfree, and Lulu Qian. A cargo-sorting DNA robot. *Science*, 357(6356):eaan6558, 2017. doi:10.1126/science.aan6558.
- 29 Anupama J. Thubagere, Chris Thachuk, Joseph Berleant, Robert F. Johnson, Diana A. Ardelean, Kevin M. Cherry, and Lulu Qian. Compiler-aided systematic construction of large-scale DNA strand displacement circuits using unpurified components. *Nature Communications*, 8:14373, 2017. doi:10.1038/ncomms14373.
- 30 Ye Tian, Yu He, Yi Chen, Peng Yin, and Chengde Mao. A DNzyme that walks processively and autonomously along a one-dimensional track. *Angewandte Chemie International Edition*, 44:4355–4358, 2005. doi:10.1002/anie.200500703.
- 31 Michael W. Trosset. Distance matrix completion by numerical optimization. *Computational Optimization and Applications*, 17:11–22, 2000. doi:10.1023/A:1008722907820.
- 32 Shelley F. J. Wickham, Jonathan Bath, Yousuke Katsuda, Masayuki Endo, Kumi Hidaka, Hiroshi Sugiyama, and Andrew J. Turberfield. A DNA-based molecular motor that can navigate a network of tracks. *Nature Nanotechnology*, 7:169–173, 2012. doi:10.1038/nnano.2011.253.
- 33 Boyan Yordanov, Jongmin Kim, Rasmus L. Petersen, Angelina Shudy, Vishwesh V. Kulkarni, and Andrew Phillips. Computational design of nucleic acid feedback control circuits. *ACS Synthetic Biology*, 3(8):600–616, 2014. doi:10.1021/sb400169s.
- 34 David Yu Zhang, Rizal F. Hariadi, Harry M. T. Choi, and Erik Winfree. Integrating DNA strand-displacement circuitry with DNA tile self-assembly. *Nature Communications*, 4:1965, 2013. doi:10.1038/ncomms2965.
- 35 David Yu Zhang and Georg Seelig. Dynamic DNA nanotechnology using strand-displacement reactions. *Nature Chemistry*, 3:103–113, 2011. doi:10.1038/nchem.957.
- 36 Fei Zhang, Shuoxing Jiang, Siyu Wu, Yulin Li, Chengde Mao, Yan Liu, and Hao Yan. Complex wireframe DNA origami nanostructures with multi-arm junction vertices. *Nature Nanotechnology*, 10:779–784, 2015. doi:10.1038/nnano.2015.162.

A Appendix

$$\frac{G = (V, \text{length}, \text{color}, A, \text{toehold}, E) \quad a \in A \setminus E \quad a \cap \text{sites}(E) = \emptyset}{G' = (V, \text{length}, \text{color}, A, \text{toehold}, (E \cup \{a\})) \quad \text{plausible}(G')} \quad (\text{BIND})$$

$$G \rightarrow G'$$

$$\frac{G = (V, \text{length}, \text{color}, A, \text{toehold}, E) \quad a = \{s_1, s_2\} \in E \quad \text{toehold}(a) \quad \neg \text{sameSpecies}(s_1, s_2, G')}{G' = (V, \text{length}, \text{color}, A, \text{toehold}, (E \setminus \{a\})) \quad \text{plausible}(G')} \quad (\text{UNBIND})$$

$$G \rightarrow G'$$

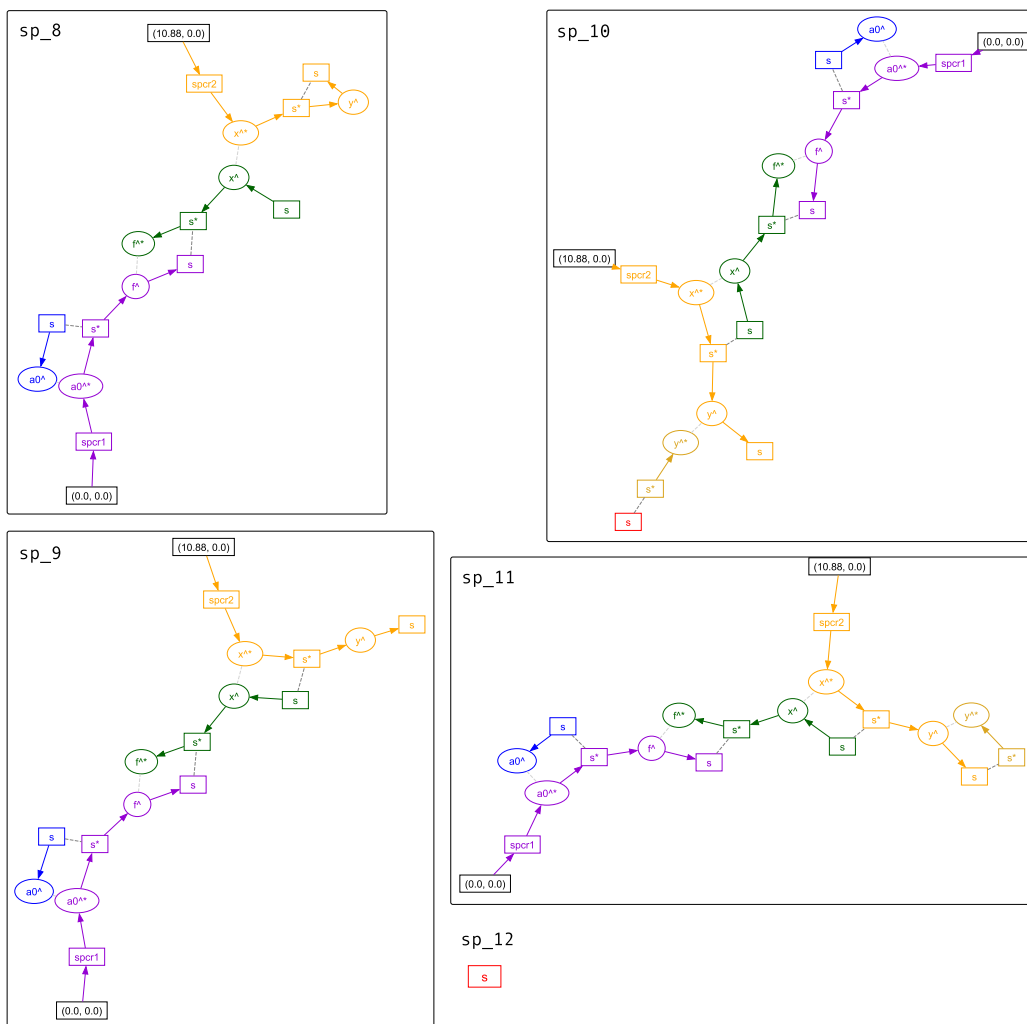
$$\frac{G = (V, \text{length}, \text{color}, A, \text{toehold}, E) \quad e = \{s, s'\} \in E \quad a = \{s, s''\} \in A \setminus E \quad s'' \notin \text{sites}(E) \quad \text{sameSpecies}(s, s'', G)}{G' = (V, \text{length}, \text{color}, A, \text{toehold}, ((E \setminus \{e\}) \cup \{a\})) \quad \text{plausible}(G')} \quad (\text{MIGRATE3})$$

$$G \rightarrow G'$$

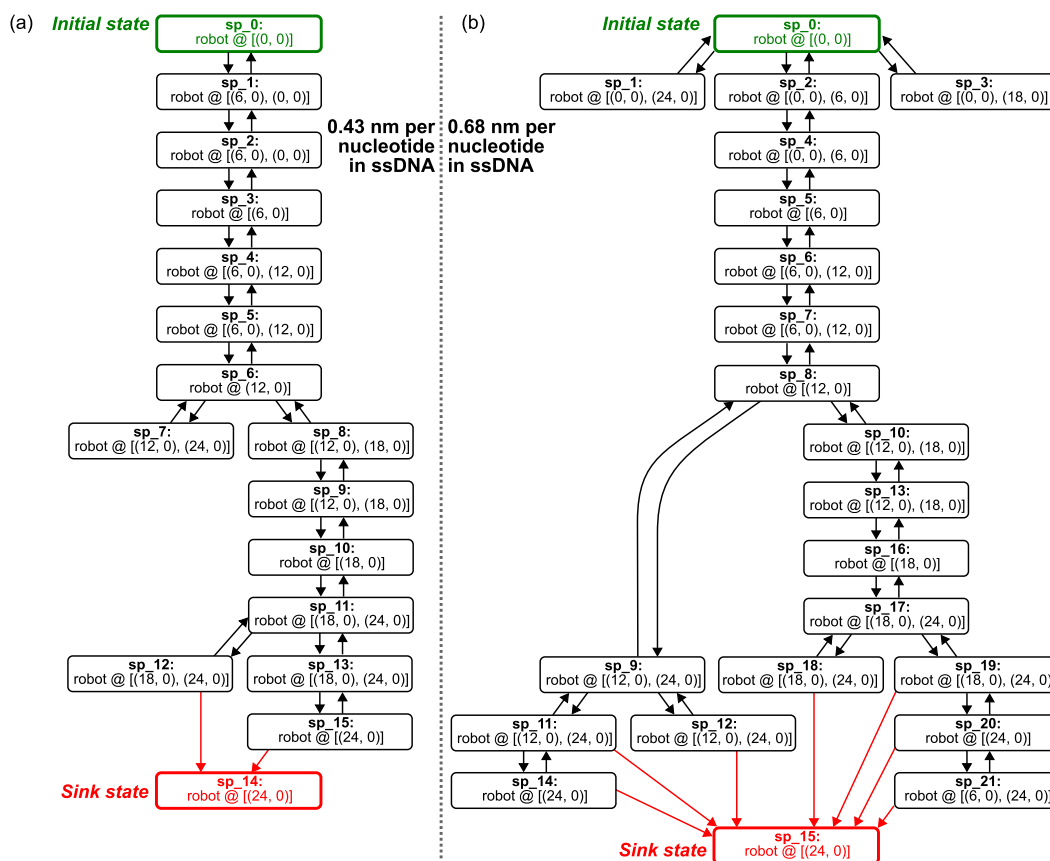
$$\frac{G = (V, \text{length}, \text{color}, A, \text{toehold}, E) \quad e_1 = \{\text{3pr}(s_4), s_1\} \in E \quad e_2 = \{\text{3pr}(s_1), s_2\} \in E \quad e_3 = \{\text{3pr}(s_2), s_3\} \in E \quad e_4 = \{\text{3pr}(s_3), s_4\} \in E}{G' = (V, \text{length}, \text{color}, A, \text{toehold}, ((E \setminus \{e_2, e_4\}) \cup \{a_1, a_2\})) \quad \text{plausible}(G')} \quad (\text{MIGRATE4})$$

$$G \rightarrow G'$$

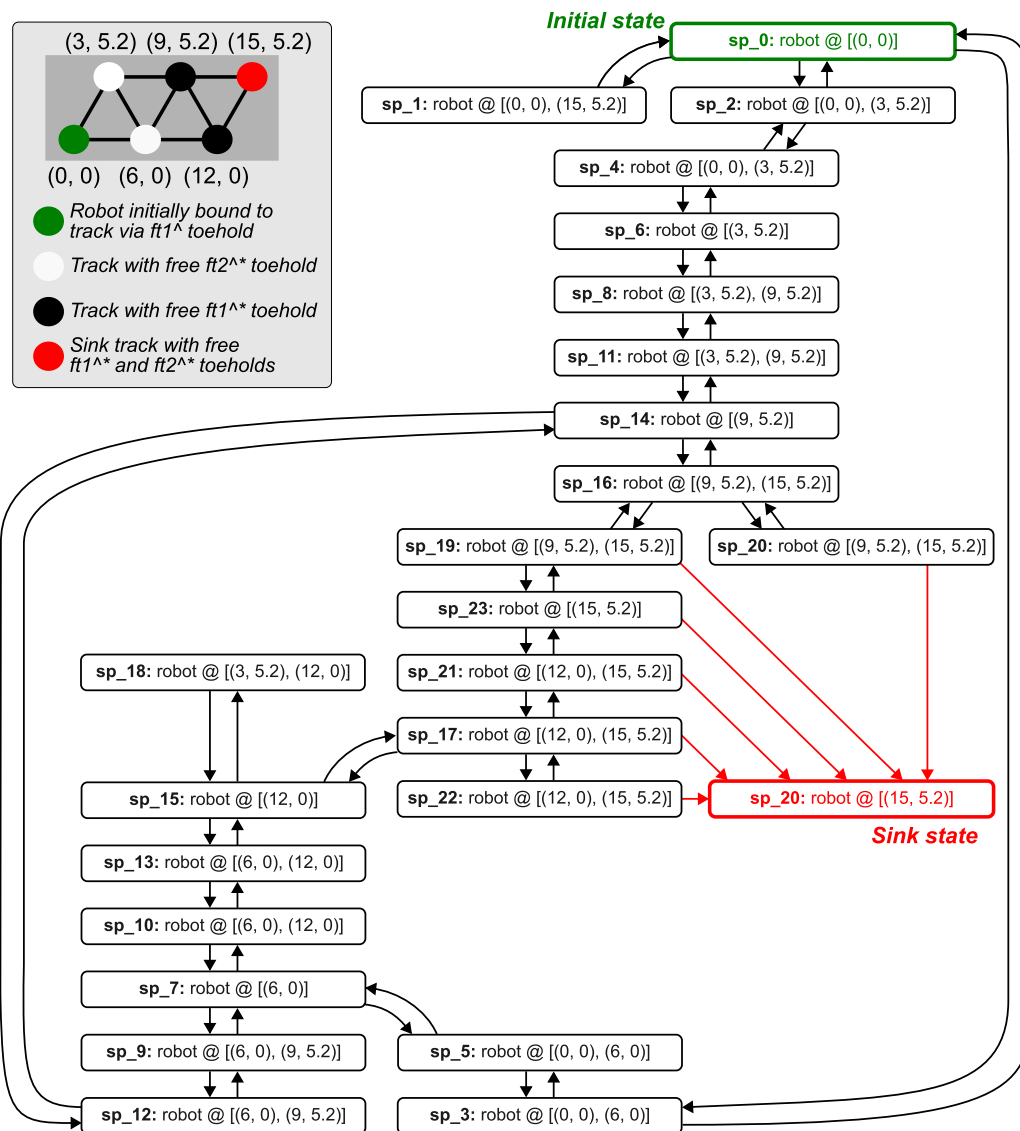
■ **Figure A.1** Inference rules defining a geometrically-aware transition relation (\rightarrow) between strand graphs; reproduced from previous work [12] for convenience. The *plausible* predicate defines the interface to a geometric constraint solver, implemented here by a randomized structure sampling algorithm, which determines if the reaction is permitted by geometry based on the geometric plausibility of the candidate product structure.



■ **Figure A.2** Part 2 of 2. Example listing of reactions enumerated for the hairpin chain reaction system of Chatterjee et al. [6]. Images generated by our prototype implementation have been manually combined in a more space-efficient manner. Species in boxes are tile species, with tether coordinates indicated. Rate constants omitted as the estimation of kinetic rates is not the focus of this work.



■ **Figure A.3** CRNs generated for random walk of the molecular robot system of Thubagere et al. [28] on a linear track of five locations spaced 6 nm apart, comparing the results obtained when the length per nucleotide for single-stranded DNA is assumed to be either (a) 0.43 nm or (b) 0.68 nm. Red arrows indicate irreversible reactions leading into the sink state where the robot is irreversibly bound to the final track site. With a larger distance assumed per nucleotide, we observe more reactions for a given track size, which makes sense because the “longer” robot strands can reach further to interact with non-adjacent track sites.



■ **Figure A.4** CRN generated for random walk of the molecular robot system of Thubagere et al. [28] on a hexagonal lattice track of locations 6 nm apart (track design illustrated in inset). Red arrows indicate irreversible reactions leading into the sink state where the robot is irreversibly bound to the final track site. As the number of tracks and complexity of the system increases, we see a corresponding increase in the number of possible transient side-reactions.