Activatable Tiles: Compact, Robust Programmable Assembly and Other Applications

Urmi Majumder, Thomas H. LaBean, and John H. Reif

Department of Computer Science,
Duke University, Durham, NC, USA
{urmim,thl,reif}@cs.duke.edu
http://www.cs.duke.edu/~reif

Abstract. While algorithmic DNA self-assembly is, in theory, capable of forming complex patterns, its experimental demonstration has been limited by significant assembly errors. In this paper we describe a novel protection/deprotection strategy to strictly enforce the direction of tiling assembly growth to ensure the robustness of the assembly process. Tiles are initially inactive, meaning that each tile's output pads are protected and cannot bind with other tiles. After other tiles bind to the tile's input pads, the tile transitions to an active state and its output pads are exposed, allowing further growth. We prove that an activatable tile set is an instance of a compact, error-resilient and self-healing tile-set. We also describe a DNA design for activatable tiles and a deprotection mechanism using DNA polymerase enzymes and strand displacement. We conclude with a discussion on some applications of activatable tiles beyond computational tiling.

Keywords: DNA-assembly, error-correction, molecular computation.

1 Introduction

The potential of self-assembling DNA nanostructures is derived from the predictable properties of DNA hybridization as well as from the assembly's theoretical power to instantiate any computable pattern [3]. Winfree [1] formalized this process of tiling assembly growth when he proposed Tile Assembly Model (TAM) which describes how a complex structure can spontaneously form from simple components called "tiles"; this assembly can also perform computation. However, the main problem for a practical implementation of TAM based assemblies is that tile additions are very error-prone. Experiments show that error rates can be as high as 1% to 8% [4,5]. The primary kind of error encountered in DNA tile assembly experiments is known as the *error by insufficient attachment* [7], which occurs when a tile violates the TAM rule stating that a tile may only be added if it binds strongly[1] enough. Thus there is a mismatch between theoretical models of DNA tiles and reality, providing major challenges in applying this model to real experiments.

[1] In the TAM for temperature $\tau = 2$, a tile binds strongly either using at least one strong bond or two weak bonds.

M.H. Garzon and H. Yan (Eds.): DNA 13, LNCS 4848, pp. 15–25, 2008.

There have been several designs of error-resilient tile sets [6,7,8] that perform "proofreading" on redundantly encoded information [8] to decrease assembly errors. Recall that the primary kinds of error in assembly experiments are: (i) growth error that occurs when a tile with one weak bond attaches at a location where a tile with two weak bonds should have been attached, (ii) facet nucleation error that occurs when a weakly binding tile attaches to a site where no tile should currently attach and (iii) spontaneous nucleation error that occurs when a large assembly grows without a seed tile. Each of these error-resilient tile sets [6,7,8], however, addresses only certain errors and proposes a construction that works with limited classes of tile sets. Additionally, most constructions result in greatly increased tile set size, hindering practical implementation. This leads to a major open question in error-resilient self-assembly: Is it possible to design a compact tile set that can address all three kinds of errors simultaneously? Our *activatable tile set* is an effort towards achieving this ultimate goal.

Limitations of Previous Approaches towards Robust Assembly: Existing error-resilient tile sets assume directional growth. This is a very strong assumption because experiments show that real tiles do not behave in such a fashion. The assumption, however, underlies the growth model in TAM. Thus, a potential solution to minimizing assembly errors is to enforce this directionality constraint. Observe that if we start with a set of "deactivated" tiles which activate in a desired order, we can enforce a directional assembly at the same scale as the original one. Such a system can be built with minimal modifications of existing DNA nanostructures [9,10,11].

Previous Approaches to direct Tiling Assembly Procedures: The snaked-proofreading technique of Chen et al. [7] provided the main inspiration for *activatable tiles*. This scheme replaces each original tile by a $k \times k$ block of tiles. The assembly process for a block doubles back on itself such that nucleation error cannot propagate without locally forcing another insufficient attachment. Can such a growth order be enforced at the original scale of the assembly? Other motivating work has been from Dirks et al. [2], who designed a system where monomer DNA nanostructures, when mixed together, do not hybridize until an initiator strand is added. Can the idea of triggered self-assembly be used in the context of computational DNA tiling?

The answers to both questions are yes. The key idea is to start with a set of "protected" DNA tiles, which we call *activatable tiles*; these tiles do not assemble until an initiator nanostructure is introduced to the solution. The initiator utilizes strand displacement to "strip" off the protective coating on the input sticky end(s) of the appropriate neighbors [12]. When the input sticky ends are completely hybridized, the output sticky ends are exposed. DNA polymerase enzyme can perform this deprotection, since it can act over long distances (e.g: across tile core) unlike strand displacement. The newly exposed output sticky ends, in turn, strip the protective layer off the next tile along the growing face of the assembly. The use of polymerase in this context is justified because of its successful use in PCR, a biochemistry technique often used for exponentially amplifying DNA. PCR has been so successful that it has several commercial

applications including genetic fingerprinting, paternity testing, hereditary disease detection, mutagenesis and more. Further PCR amplification of megabase DNA has also been done [21]. In nature most organisms copy their DNA in the same way making polymerase an excellent choice for reliable deprotection over long distances. Many repeated rounds of primer polymerization are required in conventional PCR. In contrast, we are using only a single round of primer polymerization (similar to a single round of PCR) to expose the desired sticky ends in our activatable tiles. Other proteins, such as helicase which are useful for DNA replication may be used for unwrapping our protection strand, but we have not yet investigated this direction quite thoroughly. Another important observation in this context is that although polymerase and the activatable tile are of comparable sizes, when the polymerase attaches to the primer, which is bound to the protection strand, it is only bound at the concave open face of the assembly (ensured by the sequential assembly growth) and hence there is no possibility of steric hindrance.

Enzyme-free Activated Tiles: The most relevant previous work that has been recently brought to our attention is probably that of Fujibayashi et al. [23,24]: the Protected Tile Mechanism (PTM) and the Layered Tile Mechanism (LTM) which utilize DNA protecting molecules to form kinetic barriers against spurious assembly. Although this is an enzyme-free circuit, in the PTM, the output sticky ends are not protected and thus they can bind to a growing assembly before the inputs are deprotected and hence cause an error. In the LTM, the output sticky ends are protected only by 3 nucleotides each and can be easily displaced causing the above-mentioned error. Error resilience can only be guaranteed if we can ensure a deprotection from input to output end.

Our Results and the Organization of the Paper: Section 1 introduced the notion of deprotection and discussed the need for activatable tiles in computational assemblies. Section 2 describes the abstract and kinetic models for activatable tiles that build on Winfree's original TAMs, with the primary difference being that each tile now has an associated finite state machine. In Section 3, we prove that the activatable tile set is an instance of a compact, error-resilient and self-healing tile set. In Section 4, we describe the DNA design of an example one dimensional activatable tile and its deprotection using both strand displacement and DNA polymerization. In Section 5 we discuss some applications of activatable tiles beyond computational assemblies as a concentration/sensing system and reaction catalyzation. In Section 6 we conclude the paper.

2 The Activatable Tile Assembly Models

An abstract model is a theoretical abstraction from reality that is often easier to work with conceptually as well as mathematically. Since Winfree has already established the framework for tiling assembly models with his TAM, we build our abstract Activatable Tile Assembly Model (aATAM) and the kinetic Activatable Tile Assembly Model (kATAM) discussed in this section on Winfree's abstract and kinetic TAMs respectively [1].

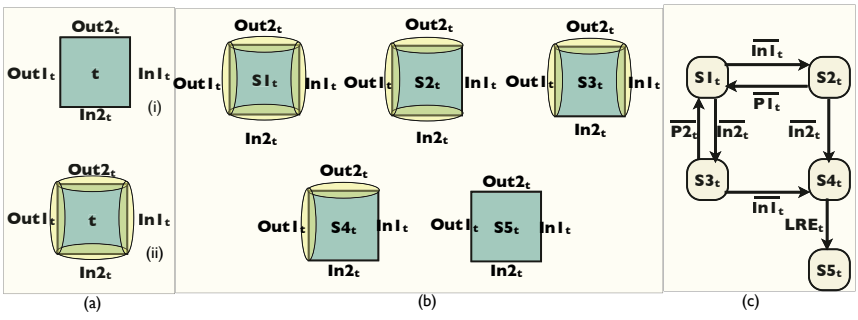

Fig. 1. (a-i) Original Abstract Rule Tile R, (a-ii) Protected version of R, (b) Different states associated with the activatable R (aR), (c) State Transition Diagram for aR. The $In1_t$ and the $In2_t$ denote the sticky ends that displaces the protections $P1_t$ and $P2_t$ from the input ends of the tile t while LRE_t is the long range effector that displaces the protection from the output end.

2.1 The Abstract Activatable Tile Assembly Model (aATAM)

The simplest version of activatable tiles starts with a set of "protected" *rule tiles*[2] that do not assemble until a pre-assembled initiator assembly, consisting of a *seed tile* and multiple *boundary tiles*, is introduced to the mixture. In the more complex version, the initiator is the seed tile alone and the boundary tiles have a protection-deprotection scheme similar to that of the rule tiles.

The aATAM is similar to the original abstract TAM (aTAM) due to Winfree [1] except that each tile type t has an associated finite state machine (FSM) M_t and hence, each tile has a state. The new abstract rule tile is shown in Figure 1(a-ii). Unlike the original tile [Figure 1(a-i)], it has all its sides protected. The states in the FSM M_t arise from the presence or absence of protection on the four sides of the tile type t (as shown in Figure 1(b)). The state transition diagram is shown in Figure 1(c).

2.2 The Kinetic Activatable Tile Assembly Model (kATAM)

The kATAM is based on Winfree's original model kTAM, but due to the the stochastic nature of the protection on all sides of the tile, additional errors need to be modeled. Therefore we need more free parameters than just r_f and $r_{r,b}$ for modeling assembly growth. Figure 2 shows the different states possible in the finite state machine for the kATAM and Figure 1(Right) shows the state transition diagram. In addition to the assumptions of kTAM, the main assumptions of kATAM are: (i) The input protection is only reversible while the output pads are still protected, (ii) Output protection is irreversible, meaning once a tile is

[2] The three main types of tiles in TAM are : (i) Rule tiles, responsible for computation in algorithmic self-assembly, (ii) Seed tile that nucleates the assembly and (iii) Boundary tiles that provide two dimensional input for computation.

completely deprotected, it cannot return to the stage where every side of the tile has a protective cover. Monomers in solution are thus either entirely protected or entirely deprotected.

The main features of the kinetic model are: (1) a tile can get knocked off the growth site after output deprotection. These unprotected tiles, however, are added to the growth site at a different rate, $r_f^{'}$, that will later be shown to be much smaller than r_f, (2) with one input match, the tile in S8 (S2) transitions to S9 (S3) at the rate of r_{dp} (deprotection) and returns to S8 (S2) at the rate of r_p (protection), (3) When both inputs are matched, the output pads (S5) are deprotected at the rate r_{dp_out}. Note that r_{dp}, r_p and r_{dp_out} are free parameters whose value depends on the experimental situation. The kinetic parameters can be derived for an example deprotection system. The description is omitted due to space constraints. Interested readers can refer to [13].

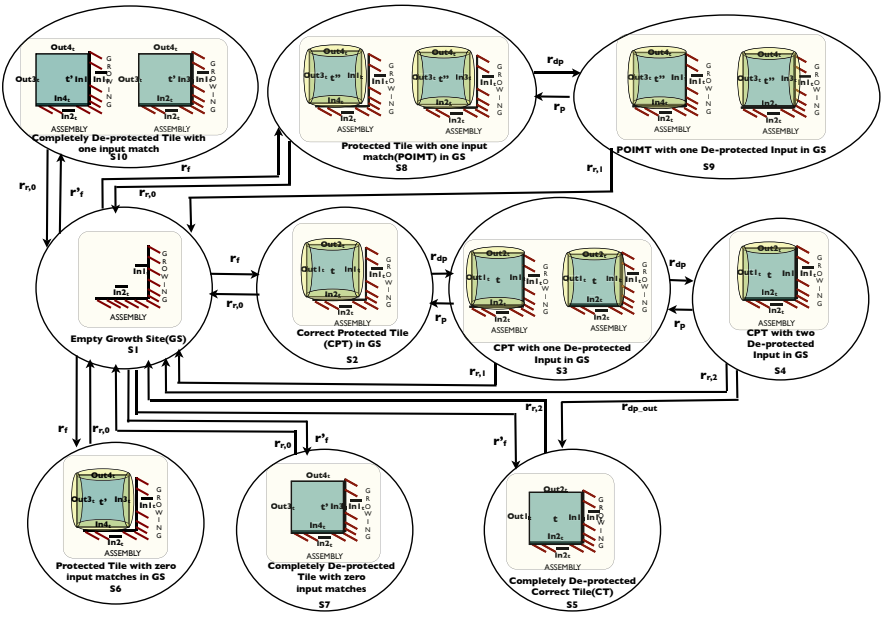

Fig. 2. State transition diagram for kATAM

Forward Rate of Erroneous Tiles: Since there are many free parameters in the kinetic model, such as $r_f, r_{r,b}, r_p$ and others we decrease the dimensionality of the parameter space by combining some of the parameters together e.g. r_p, r_{dp} and r_{dp_out}. This is done by computing the rate at which tiles become completely deprotected after reaching a growth site, thus neglecting the intermediate states in Figure 2. This new rate corresponds to the rate at which a tile reaches state $S5$ if it is in $S1$. We call this rate r_{eff} and assume that r_{eff} is a function of G_{se} such that $r_{eff} = k_f e^{(-2+\epsilon_1)G_{se}}$, where ϵ_1 is a constant between 0 and 1. Note

that r_{eff} is similar to r_f in the original kTAM. Based on the continuous time Markov Chain (CTMC) in Figure 2, we can evaluate r_{eff} as

$$r_{eff} = r_f \frac{r_{dp}}{(r_{dp} + r_{r,0})} \frac{r_{dp}}{(r_p + r_{dp} + r_{r,1})} \frac{r_{dp\_out}}{(r_{dp\_out} + r_{r,2} + r_p)}. \quad (1)$$

One primary assumptions in the model are

$$r_{r,1} > r_f > r_{eff} > r_{r,2} \text{ and}$$
$$r_{r,1} = e^{-G_{se}}, r_{r,2} = e^{-2G_{se}}, r_{eff} = e^{(-2+\epsilon_1)G_{se}}, r_f = e^{(-2+\epsilon_1+\epsilon_2)G_{se}}$$
$$\text{for some } 0 < \epsilon_1, \epsilon_2 < 1. \quad (2)$$

For simplicity of the model, we can ensure that $\epsilon_2 \ll \epsilon_1$ by adjusting the kinetic parameters in the deprotection system (e.g. toehold length in the strand displacement events, nucleotide concentration and template length for polymerization etc). Hence $r_{eff} \gg r_{r,2}$. Another important assumption we make is that DNA polymerization is irreversible and, hence, at equilibrium every tile is completely deprotected.

Based on these assumptions we can first obtain the expected fraction of completely deprotected tiles that leaves $S5$ as $\frac{r_{r,2}}{r_{r,2}+r^*} e^{-G_{mc}}$ and hence derive $r_f^{'}$, the forward rate of erroneous tiles as $e^{(-2+\epsilon_2)G_{se}}$.

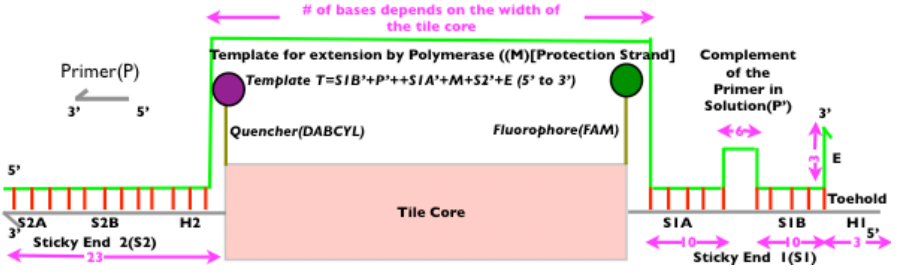

Fig. 3. Protection Strategy for a DNA Tile

3 Compact Proofreading with Activatable Tiles

Activatable tiles provide error-resilience to a growing assembly by enforcing directional growth. Ideally the output ends are never available until the corresponding input ends are completely hybridized, thus preventing both errors by insufficient attachment as well as nucleation errors. There is a small probability, however, of errors by insufficient attachment caused by tiles that leave a growth site after output deprotection. Furthermore, the computation still occurs at the original scale, unlike Chen's snaked proofreading technique [7] which increases the lattice size by a multiplicative factor of k^2. Hence, activatable tiles indeed

provide compact error-resilience. Since the seed is the only completely unprotected tile when the assembly begins and the concentration of completely unprotected rule or boundary tiles existing in solution at any given time is very low, activatable tiles can also prevent spontaneous nucleation and enforce "controlled growth".[3] We can formally prove that activatable tiles are indeed an instance of compact proofreading technique. Soloveichik et al. gave a concise definition of compact proofreading [14] and we adapt it to our ATAM:

Definition 1. *Given a small constant* $0 < q < 1$*, a sequence of deterministic tile systems* $\{T_1, T_2, T_3, \ldots\}$ *is a compact proofreading scheme for pattern* P *if (i)* T_N *produces the full infinite pattern* P *under the aATAM, (ii)* T_N *has* $poly(\log N)$ *tile types (*$poly(n)$ *denotes* $n^{O(1)}$*) and (iii)* T_N *produces the correct* $N \times N$ *initial portion of the pattern* P *with probability at least* q *in time* $O(Npoly(logN))$ *in the kATAM for some values of the free parameters in the model.*

Theorem 1. *The activatable Tile System* A_N *is a compact proofreading scheme.*

Proof. Let the tile system in aTAM be T_N and the activatable tile system be A_N. A_N is the same as T_N except that each tile type has an associated FSM. Since in aATAM activatable tiles can bind to a growth site only if they can bind strongly enough (just as in aTAM), A_N can produce the whole system correctly under aATAM so the first condition is satisfied. Moreover, $|A_N| = |T_N|$, the only difference being that we start the assembly with a "protected" version of T_N. Since this work is concerned with only deterministic tile systems, the argument of Soloveichik et al. [14] applies and we need only a constant number of tile types so long the tile set has a locally deterministic assembly sequence.

The argument for the third condition is similar to that of Chen et al. [7]. In this model, errors are only caused by insufficient attachments; these errors are caused by tiles dissociating from growth sites after their output protection has been stripped off. In an insufficient attachment event, first an unprotected monomer (with a single binding site match) attaches at the rate of $r_f^{'}$. However, before this tile is knocked off at the rate of $r_{r,1}$, a second tile (protected/unprotected) can attach to the first tile at the rate $r_f^{'} + r_{eff}$. Thus, based on the corresponding CTMC we can say that the rate of an insufficient attachment is

$$r_{insuf} = \frac{r_f^{'}(r_f^{'} + r_{eff})}{r_{r,1} + r_f^{'} + r_{eff}} = e^{(-3+\epsilon_1+\epsilon_2)G_{se}} \frac{1 + e^{-(\epsilon_1-\epsilon_2)G_{se}}}{1 + e^{-(1-\epsilon_1)G_{se}} + e^{-(1-\epsilon_2)G_{se}}} \quad (3)$$

Our goal with respect to a particular growth site is to bury the correct tile k levels deep before an insufficient attachment event occurs.[4] In other words, if we have a $k \times k$ square whose left bottom corner location is occupied by this tile, then the $k \times k$ square completes before an insufficient attachment event occurs. This puts the tile under consideration into a "k-frozen" state. The process of

[3] Controlled growth is defined to be the growth occurring for parameter values in a certain part of the kinetic parameter space, such that (i) growth does occur, (ii) errors are rare and (iii) growth not seeded by the seed tile is rare [15].

[4] The time taken for single tile attachment is $O(1/r_{eff})$ which is less than $1/r_{insuf}$.

tile attaching or detaching in a 2D assembly can be modeled as a random walk.[5] Note that the forward growth (tile association at the output ends of the current tile) happens at the rate of $r_{eff} + r_f^{'}$ while the backward growth (dissociation of the current tile) has a rate of $r_{r,2}$. Thus, the average rate of growth (the mean of forward and backward rates) $\overline{r}$ is $\frac{1}{2}(r_{eff} + r_f^{'} + r_{r,2})$ and the expected time taken for this $k \times k$ square to grow is $O(k^4/\overline{r})$ since in a 2D random walk, we have to take k^4 steps in expectation in order to cover k^2 locations.

Thus, for any small ϵ_{insuf}, one can find a constant c_{insuf} such that, with probability $1-\epsilon_{insuf}$, no insufficient attachment happens at this specific location but a correct tile becomes k-frozen within time $O(k^4/\overline{r})$. In other words, $(k^4/\overline{r}) < (c_{insuf}/r_{insuf})$. Hence, for a given k, such that with high probability a given growth site is filled correctly and buried k levels deep in $O(k^4/\overline{r})$ time. For constant kinetic parameters and k, this time is also constant. Hence we can use the same argument as Adleman et al. [19] and show that the $N \times N$ square is completed in expected $O(N)$ time. □

Compact Self-healing with Activatable Tiles: The impact of activatable tiles goes beyond the compact error-resilience which is a primary concern for fault tolerant self-assembly. In case of gross external damage, e.g. a hole created in a growing tiling assembly, activatable tiles can repair the damage with minimal error by enforcing directional growth. Since the original, self-assembled lattice was formed by algorithmic accretion in the forward direction, only forward regrowth is capable of rebuilding the correct structure. The protected monomers in the solution ensure a forward directional accretion. There is a small probability, however, of backward growth from the unprotected monomers that were once part of the original tiling assembly and dissociated after outputs are deprotected. The likelihood is comparatively small since the forward reaction rate depends on concentration of the monomers and the protected tiles are much more abundant than their unprotected counterparts. Defining size in terms of number of tiles, we conclude the following theorem:

Theorem 2. *With high probability, a damaged hole of size S (small compared to the assembly size) is repaired in time $O(S^2)$, for suitable kATAM parameters.*

Proof. Observe that the maximum rate of error due to backward growth is bounded by $r_f^{'}$ while the forward rate of growth is $r_{eff} + r_f^{'}$. Observe that $\overline{r} > r_f^{'}$. Using the same technique as in Theorem 1, we can prove that the hole can be repaired in $O(S^2/\overline{r})$ by a 2D random walk on the set of S tile positions on the 2D plane. We can further argue that for any small ϵ_{heal} $(0 < \epsilon_{heal} < 1)$, one can find a constant c_{heal} such that with probability $1 - \epsilon_{heal}$, $(S^2/\overline{r}) < (c_{heal}/r_f^{'})$. For a given S, we can compute G_{se} so that there is no backward growth when a hole of size at most S gets repaired in $O(S^2)$ time assuming constant parameters. □

[5] The stochastic process of tile attachment and detachment in self -assembly has often been modeled as a random walk [7]. Further this is similar to the lattice gas model where modeling interactions as random walks is quite well established.

4 DNA Design of One Dimensional Activatable Tiles

The DNA design of one dimensional (1D) activatable tiles is very helpful in understanding the more complex DNA design of two dimensional (2D) activatable tiles. It is also motivated by the need for a protection strategy for tiles that self-assemble into a 1D lattice, such as the boundary of the computational tiling. Hence we first describe the DNA design of a 1D activatable tile. Figure 3 gives the sequence design of a 1D activated tile. Some of the key features of the tile design are: (i) The sticky ends are protected by the protection strand M, (ii) For adjacent tiles, the protection strand needs to be arranged in a different manner so as to satisfy both constraints on the direction for sticky end matching as well as the template for polymerization (not shown here), resulting in two kinds of tile types, (iii) The 3 base portion (E) at the $3'$ end of the protection strand in the tile design prevents polymerization of the toehold $H1$, (iv) The portion of the protection strand which hybridizes to the primer P is held tightly in a hairpin loop of six bases between two subportions of the input sticky end, (v) The fluorophore and the quencher are positioned such that the flourophore is quenched only when correct tiles hybridize.

How does an activated tile deprotect its neighbor? The idea is quite simple: the toehold H on the input sticky end $S1$ of the protected tile (say Tile 1) is used to displace the protection strand M on it; after the input sticky end of the Tile 1 and the output sticky end of the deprotected tile (say Tile 2) are completely hybridized, the protection strand M is freed from the input end of Tile 1; the primer P can now attach to the complementary portion $P^{'}$ on the protection strand M that was earlier held tightly in a hairpin loop. Polymerase next binds to the $3^{'}$ end of the primer and extends it to the output end of Tile 1. Eventually, the output sticky end of Tile 1 is exposed.

Our DNA design for 2D activatable tiles is a direct extension of our 1D activatable tiles. Interested readers can refer to [13].

5 Other Applications of Activatable Tiles

Beyond the applications to computational tiling, activatable tiles can also be used as a novel system for sensing and concentration. For example, one can design a modified activatable tile to include a docking site for a specific target molecule. Initially, the tiles are in the inactive state; they are neither bound to a target molecule nor they are assembled together. When a target molecule binds to the tile's docking site, the tile transitions from an inactive to an active state. Tiles in the active state can assemble. As the activated tiles assemble, the target molecules are concentrated making an excellent concentration system.

Activatable tiles can also be used for reaction catalyzation. Suppose that for some small k, the goal is to gather k distinct types of target molecules to initiate or catalyze a chemical reaction. Just as with the sensing system, one can design k distinct activated tiles, each with a docking site for a different target molecule. These tiles become active only when they are carrying their target molecules.

Once activated, these k distinct tiles assemble into a small tiling lattice, putting the target molecules in close proximity, and allowing them to react. Additionally, the reaction products can be used to disassemble the lattices and deactivate the tiles, allowing them to be reused. Observe that the binding site on the same face of each tile type is so designed that after assembly, the molecules bound to the tiles will be close to each other. They are never bound inside the lattice and therefore, the reaction can never become slower. Although this is quite a novel idea, the concept of DNA directed chemistry has been explored quite extensively in the recent years (See [22]).

6 Conclusion

In spite of the fact that it may be impossible to eliminate errors completely from the assembly process, activatable tiles appear to be quite promising. Thus, as a part of future work, we not only intend to have an experimental validation, but also evaluate our deprotection strategy with computer simulation, particularly compare it with the simulation results from Fujibayashi et al.'s enzyme-free activated tile model [23,24]. We conclude with one interesting open question: Can combining overlay redundancy techniques [16] with the idea of activatable tiles further improve the compact error-resilience of self-assembly experiments?

Acknowledgments. This work is supported by NSF ITR Grants EIA-0086015 and CCR-0326157, NSF QuBIC Grants EIA-0218376 and EIA-0218359, and DARPA/AFSOR Contract F30602-01-2-0561. The authors thank Erik Winfree for bringing the work of Satoshi Murata to our attention. Majumder also thanks Erik Halvorson for useful edits and discussion.

References

1. Winfree, E.: Algorithmic Self-Assembly of DNA Caltech (1998)
2. Dirks, R.M., Pierce, N.A.: PNAS 101(43), 15275–15278 (2004)
3. Wang, H.: Bell System Tech Journal 40(1), 1–41 (1961)
4. Barish, R.D., Rothemund, P.W.K., Winfree, E.: Nano Letters 5(12), 2586–2592
5. Rothemund, P.W.K., Papadakis, N., Winfree, E.: PLoS Biology 2 (12), 424 (2004)
6. Reif, J.H., Sahu, S., Yin, P.: DNA10. In: Ferretti, C., Mauri, G., Zandron, C. (eds.) DNA Computing. LNCS, vol. 3384, pp. 293–307. Springer, Heidelberg (2005)
7. Chen, H.L., Goel, A.: DNA 10. In: Ferretti, C., Mauri, G., Zandron, C. (eds.) DNA Computing. LNCS, vol. 3384, pp. 62–75. Springer, Heidelberg (2005)
8. Winfree, E., Bekbolatov, R.: DNA 9. In: Chen, J., Reif, J.H. (eds.) DNA Computing. LNCS, vol. 2943, pp. 126–144. Springer, Heidelberg (2004)
9. Yan, H., Park, S.H., Finkelstein, G., Reif, J.H., LaBean, T.H.: Science 301(5641), 1882–1884 (2003)
10. LaBean, T.H., Yan, H., Kopatsch, J., Liu, F., Winfree, E., Reif, J.H., Seeman, N.C.: J. Am. Chem. Soc. 122, 1848–1860 (2000)
11. Winfree, E., Liu, F., Wenzler, L.A., Seeman, N.C.: Nature 394 (1998)
12. Thompson, B.J., Camien, M.N., Warner, R.C.: PNAS 73(7), 2299–2303 (July 1976)

13. http://www.cs.duke.edu/~reif/paper/urmi/activatable/activatable.pdf
14. Soloveichik, D., Winfree, E.: SICOMP 36(6), 1544–1569 (2007)
15. Winfree, E.: Simulations of Computing by Self-Assembly Caltech CS Report, 22 (1998)
16. Sahu, S., Reif, J.H.: DNA 12. In: Mao, C., Yokomori, T. (eds.) DNA Computing. LNCS, vol. 4287, pp. 223–238. Springer, Heidelberg (2006)
17. Saturno, J., Blanco, L., Salas, M., Esterban, J.A: J. Bio. Chem. 270(52), 31235–31243 (1995)
18. Thompson, B.J., Escarmis, C., Parker, B., Slater, W.C., Doniger, J., Tessman, I., Warner, R.C.: J. Mol. Biol. 91, 409–419 (1975)
19. Adleman, L., Cheng, Q., Goel, A., Huang, M.D.: Proceedings of STOC. pp. 740–748 (2001)
20. Schulman, R., Winfree, E.: DNA 10. In: Ferretti, C., Mauri, G., Zandron, C. (eds.) DNA Computing. LNCS, vol. 3384, pp. 319–328. Springer, Heidelberg (2005)
21. Grothues, D., Cantor, C.R., Smith, C.L.: Nuc. Acid. Res. 21(5), 1321–1322 (1993)
22. Rosenbaum, D.M., Liu, D.R.: J. Am. Chem. Soc. 125, 13924–13925 (2003)
23. Fujibayashi, K., Murata, S.: DNA 10. In: Ferretti, C., Mauri, G., Zandron, C. (eds.) DNA Computing. LNCS, vol. 3384, pp. 113–127. Springer, Heidelberg (2005)
24. Fujibayashi, K., Zhang, D., Winfree, E., Murata, S.: In submission