Design, Simulation, and Experimental Demonstration of Self-assembled DNA Nanostructures and Motors

John H. Reif, Thomas H. LaBean, Sudheer Sahu,
Hao Yan, and Peng Yin

Department of Computer Science, Duke University,
Box 90129, Durham, NC 27708-0129, USA
{reif, thl, sudheer, hy1, py}@cs.duke.edu

Abstract. Self-assembly is the spontaneous self-ordering of substructures into superstructures, driven by the selective affinity of the substructures. Complementarity of DNA bases renders DNA an ideal material for programmable self-assembly of nanostructures. DNA self-assembly is the most advanced and versatile system that has been experimentally demonstrated for programmable construction of patterned systems on the molecular scale. The methodology of DNA self-assembly begins with the synthesis of single strand DNA molecules that self-assemble into macromolecular building blocks called DNA tiles. These tiles have single strand "sticky ends" that complement the sticky ends of other DNA tiles, facilitating further assembly into larger structures known as DNA tiling lattices. In principle, DNA tiling assemblies can form any computable two or three-dimensional pattern, however complex, with the appropriate choice of the tiles' component DNA. Two-dimensional DNA tiling lattices composed of hundreds of thousands of tiles have been demonstrated experimentally. These assemblies can be used as programmable scaffolding to position molecular electronics and robotics components with precision and specificity, facilitating fabrication of complex nanoscale devices. We overview the evolution of DNA self-assembly techniques from pure theory, through simulation and design, and then to experimental practice. In particular, we begin with an overview of theoretical models and algorithms for DNA lattice self-assembly. Then we describe our software for the simulation and design of DNA tiling assemblies and DNA nano-mechanical devices. As an example, we discuss models, algorithms, and computer simulations for the key problem of error control in DNA lattice self-assembly. We then briefly discuss our laboratory demonstrations of DNA lattices and motors, including those using the designs aided by our software. These experimental demonstrations of DNA self-assemblies include the assembly of patterned objects at the molecular scale, the execution of molecular computations, and the autonomous DNA walking and computing devices.

1   Introduction

Self-assembly is the spontaneous self-ordering of substructures into superstructures driven by the selective affinity of the substructures. This paper focuses on a method for self-assembly known as *DNA self-assembly*, where DNA provides a molecular scale material for effecting this programmable self-assembly, using the selective affinity of

pairs of DNA strands to form DNA nanostructures. Self-assembling nanostructures composed of DNA molecules offer great potential for bottom-up nanofabrication of materials and objects with smaller features than ever previously possible [13, 30, 37]. The methodology of DNA self-assembly begins with the synthesis of single-strand DNA molecules that self-assemble into macromolecular building blocks called DNA tiles. These tiles have sticky ends that match the sticky ends of other DNA tiles, facilitating further assembly into larger structures known as DNA tiling lattices. In principle, DNA tiling assemblies can be made to form any computable two- or three-dimensional pattern, however complex, with the appropriate choice of the tiles' component DNA.

DNA self-assembly is an emerging subfield of nanoscience with the development of its theoretical basis and a number of moderate to large-scale experimental demonstrations. Recent experimental results indicate that this technique is scalable. Periodic 2D DNA lattices have been successfully constructed with a variety of DNA tiles [15, 23, 52, 56]. These lattices are composed of up to hundreds of thousands of tiles. Molecular imaging devices such as atomic force microscopes and transmission electron microscopes allow visualization of these self-assembled two-dimensional DNA tiling lattices. These assemblies can be used as scaffolding on which to position molecular electronics and other components such as molecular sensors with precision and specificity. The programmability lets this scaffolding have the patterning required for fabricating complex devices made of these components. Potential applications of DNA self-assembly and scaffolding include nanoelectronics, biosensors, and programmable/autonomous molecular machines.

In addition to manufacturing DNA lattices, DNA has also been demonstrated to be a useful material for molecular computing systems [1, 3, 6, 21, 22] and mechanical devices [19, 24, 57, 63]. In particular, the self-assembly of DNA tiles can also be used as a powerful computational mechanism [16, 27, 47, 49], which in theory holds universal computing power [53]. See [32] for a more detailed survey of current experimental work in self-assembled DNA nanostructures. Also, see [26] and [30] for comprehensive surveys of the larger field of DNA computation (also known as biomolecular computation).

In this paper, we overview the evolution of DNA self-assembly techniques from pure theory, through simulation and design, and then to experimental practice. The rest of the paper is organized as follows. In Section 2, we overview the theoretical work in self-assembly. In Section 3, we describe software for the simulation and design of DNA nanostructures and motors. As a concrete example, in Section 4 we discuss error control, which we feel is a major theoretical and practical challenge remaining in the area of DNA self-assembly. Finally, in Section 5 we give a discussion of experimental practice in DNA nanostructures.

2   The Theory of Self-assembly

This section overviews the emerging theory of self-assembly.

Domino Tiling Problems. The theoretical basis for self-assembly has its roots in *Domino Tiling Problems* (also known as Wang tilings) as defined by Wang [45]. For comprehensive text, see [10]. The input is a finite set of unit size square tiles. The sides of each square are labeled with symbols over a finite alphabet. Additional restrictions

may include the initial placement of a subset of the these tiles, and the dimensions of the region where tiles must be placed. Assuming an arbitrarily large supply of each tile, the problem is to place the tiles, without rotation (a criterion that cannot apply to physical tiles), to completely fill the given region so that each pair of abutting tiles have identical symbols on their contacting sides.

Turing-universal and NP-Complete Self-assemblies. Domino tiling problems over an infinite domain with only a constant number of tiles were first proved by Berger to be undecidable [7]. This and subsequent proofs [7, 33] rely on constructions where tiling patterns simulate single-tape Turing machines or cellular arrays [53]. Winfree later showed that computation by self-assembly is Turing-universal [53] and so tiling self-assemblies can theoretically provide arbitrarily complex assemblies even with a constant number of distinct tile types. Winfree also demonstrated various families of assemblies which can be viewed as computing languages from families of the Chomsky hierarchy [47]. It has been proved that Domino tiling problems over polynomial-size regions are **NP**-complete [17]. Subsequently, [47], [11, 12], and [16] proposed the use of self-assembly processes (in the context of DNA tiling and nanostructures) to solve **NP**-complete combinatorial search problems such as SAT and graph coloring.

Program-size Complexity of Tiling Self-assemblies. The programming of tiling assemblies is determined simply by the set of tiles, their pads, and sometimes the choice of the initial seed tile (a special tile from which the growth of the assembly starts). A basic issue is the number of distinct tile types required to produce a specified tile assembly. The *program size complexity* of a specified tiling is the number of distinct tiles (with replacement) to produce it. Rothemund and Winfree showed that the assembly of an $n \times n$ size square can be done using $\Theta(\log n / \log \log n)$ distinct tiles and that the largest square uniquely produced by a tiling of a given number of distinct tiles grows faster than any computable function [34]. Adleman recently gave program size complexity bounds for tree shaped assemblies [3].

Massively Parallel Computation by Tiling. Parallelism reveals itself in many ways in computation by self-assembly. Each superstructure may contain information representing a different calculation (*global parallelism*). Due to the extremely small size of DNA strands, as many as 10^{18} DNA tiling assemblies may be made simultaneously in a small test tube. Growth on each individual superstructure may also occur at many locations simultaneously via *local parallelism*. The *depth* of a tiling superstructure is the maximum number of self-assembly reactions experienced by any substructure (the depth of the graph of reaction events), and the *size* of a superstructure is the number of tiles it contains. Likewise we can define the number of layers for a superstructure. For example, a superstructure consisting of an array of $n \times m$ tiles, where $n > m$ has m layers. Tiling systems with low depth, small size, and few layers are considered desirable, motivating the search for efficient computations performed by such systems. Reif was the first to consider the parallel depth complexity of tiling assemblies and gave DNA self-assemblies of linear size and logarithmic depth for a number of fundamental problems (e.g., prefix computation, finite state automata simulation, and string fingerprinting, etc.) that form the basis for the design of many parallel algorithms [27]. Furthermore, [27] showed that these elementary operations can be combined to perform

more complex computations, such as bitonic sorting and general circuit evaluation with polylog depth assemblies.

Linear Self-assemblies. Tiling systems that produce only superstructures with k layers, for some constant k, are said to use *linear self-assembly*. [27] gave some simple linear tiling self-assemblies for integer addition as well as related operations (e.g., prefix XOR summing of n Boolean bits). Seeman's group demonstrated the first example of DNA computation using DNA tiling self-assembly [22], as described in Section 5. These linear tilings were refined in [51] to a class of String tilings that have been the basis for further DNA tiling experiments in [54] described in Section 5.

Kinetic Models of Tiling Self-assembly Processes. Domino tiling problems do not presume or require a specific process for tiling. Winfree first observed that self-assembly processes can be used for computation via the construction of DNA tiling lattices [46]. The sides of the tiles are assumed to have some methodology for selective affinity, which we call *pads*. Pads function as programmable binding domains, which hold together the tiles. Each pair of pads have specified binding strengths. The self-assembly process is initiated by a singleton tile (the *seed tile*) and proceeds by tiles binding together at their pads to form aggregates known as *tiling assemblies*. The preferential matching of tile pads facilitates the further assembly into tiling assemblies. Using the kinetic modeling techniques of physical chemistry, Winfree developed a kinetic model for the self-assembly of DNA tiles [48]. Following the classical literature of models for crystal assembly processes, Winfree considers assembly processes where the tiling assembly is only augmented by single tiles (known in crystallography as *monomers*) which bind to the assembly at their tile pads [46]. The likelihood of a particular tile binding at (or dissociating from) a particular site of the assembly is assumed to be a fixed probability dependent on that tile's concentration, the respective pad's binding affinity, and a temperature parameter. In addition, Adleman developed stochastic differential equation models for self-assembly of tiles and determined equilibrium probability distributions and convergence rates for some 1-dimensional self-assemblies [2, 4]. His model allowed for binding between subassemblies and assumed a fixed probability for tile binding events independent of the size of tile assemblies. Since the movement of tile assemblies may depend on their size (and thus mass), this model might in the future be refined to make the probability for tile binding events dependent on the size of tile assemblies.

Optimization of Tiling Assembly Processes. There are various techniques that may promote assembly processes in practice. One important technique is the tuning of the parameters (tile concentration, temperature, etc.) governing the kinetics of the process. Adleman considers the problem of determining tile concentrations for given assemblies and conjectures this problem is $\sharp\mathbf{P}$-complete [3]. Various other techniques may improve convergence rates to the intended assembly. A blockage of tiling assembly process can occur if an incorrect tile binds in an unintended location of the assembly. While such a tile may be dislodged by the kinetics of subsequent time steps, it still may slow down the convergence rate of the tiling assembly process to the intended final assembly. To reduce the possibility of blockages of tiling assembly processes, Reif proposed the use of distinct tile pads for distinct time steps during the assembly [27]. [27] also described the use of self-assembled tiling *nano-frames* to constrain the region of the tiling assemblies.

3   Simulation and Design Software

Software for Kinetic Simulation of Tiling Assembly Processes. Winfree developed software for discrete time simulation of the tiling assembly processes, using approximate probabilities for the insertion or removal of individual tiles from the assembly [48]. These simulations gave an approximation to the kinetics of self-assembly chemistry and provided some validation of the feasibility of tiling self-assembly processes. Using this software as a basis, our group developed an improved simulation software package (sped up by use of an improved method for computing on/off likelihood suggested by Winfree) with a Java interface for a number of example tilings, such as string tilings for integer addition and XOR computations. In spite of an extensive literature on the kinetics of the assembly of regular crystalline lattices, the fundamental thermodynamic and kinetic aspects of self-assembly of tiling assemblies are still not yet well understood. For example, the effect of distinct tile concentrations and different relative numbers of tiles is not yet known; probably it will require an application of Le Chatelier's principle.

Software for Kinetic Simulation of Nanomechanical Devices. We have developed a software to simulate autonomous nanomechanical DNA devices driven by ligase and restriction enzymes in a solution system. This software does discrete time simulation of the ligation and restriction events on the DNA duplex fragments of the nanomechanical device. The approximate probabilities of ligation is calculated based on the concentrations of individual DNA fragments present in the solution system. These simulations can provide insight to the kinetics of such nanomechanical systems. We have used this software to simulate a DNA walker and a universal DNA Turing machine.

Software for Design of DNA Lattices and Nanomechanical Devices. A major computational challenge in constructing DNA objects is to optimize the selection of DNA sequences so that the DNA strands can correctly assemble into desired DNA secondary structures. A commonly used software package, *Sequin*, was developed by Seeman, which uses the symmetry minimization algorithm [35]. Sequin, though very useful, only provides a text-line interface and generally requires the user to step through the entire sequence selection process. Our lab recently developed a software package, *TileSoft*, which exploits an evolution algorithm and fully automates the sequence selection process [58]. TileSoft also provides the user with a graphical user interface, on which DNA secondary structure and accompanying design constraints can be directly specified and the optimized sequence information can be pictorially displayed. TileSoft is initially designed to solve optimization problem for a set of multiple tiles, but can also be used to design individual DNA objects, such as DNA nanomechanical devices.

4   Error Control in DNA Tiling Assemblies

A chief challenge in DNA tiling self-assemblies is the control of assembly errors. This is particularly relevant to computational self-assemblies, which, with complex patterning at the molecular scale, are prone to a quite high rate of error, ranging from approximately between 0.5% to 5%, and the key barrier to large-scale experimental implementation of 2D computational DNA tilings exhibiting patterning is this significant error

rate in the self-assembly process. The limitation and/or elimination of these errors in self-assembly is perhaps the single most important major challenge to nanostructure self-assembly.

There are a number of possible methods to decrease errors in DNA tilings:

(a) Annealing Temperature Optimization. This is a well known technique used in hybridization and also crystallization experiments. It can be used to decrease the defect rates at the expense of increased overall annealing time duration. In the context of DNA tiling lattices, the parameters for the temperature variation that minimize defects have not yet been determined.

(b) Error Control by Step-wise Assembly. Reif suggested the use of serial self-assembly to decrease errors in self-assembly [26].

(c) Error Control by Redundancy. There are a number of ways to introduce redundancy into a computational tiling assembly. In [31] we describe a simple method that can be developed for linear tiling assemblies: we replace each tile with a stack of three tiles executing the same function, and then add additional tiles that essentially 'vote' on the pad associations associated with these redundant tiles. This results in a tiling of increased complexity but still linear size. This error resistant design can easily be applied to the integer addition linear tiling described above, and similar redundancy methods may be applied to higher dimension tilings.

Work in 2003 by Winfree provided a method to decrease tiling self-assembly errors without decreasing the intrinsic error rate of assembling a single tile, however, his technique resulted in a final assembled structure that is four times the size of the original one [50].

Recently we have developed improved methods for compact error-resilient self-assembly of DNA tiling assemblies and analyzed them by probabilistic analysis, kinetic analysis, and computer simulation [29]; and plan to demonstrate these error-resilient self-assembly methods by a series of laboratory experiments. Our compact error-resilient tiling methods do not increase the size of the tiling assembly. They use 2-way overlay redundancy such that a single pad mismatch between a tile and its immediate neighbor forces at least one further pad mismatch between a pair of adjacent tiles in the neighborhood of this tile. Theoretical probabilistic analysis and empirical studies of the computer simulation of Sierpinsky Triangle tilings have been used to validate these error-resilient 2-way overlay redundancy tiling results; the analysis shows that the error rate is considerably reduced.

5   Experimental Progress

DNA Hybridization. Single strand DNA is a polymer that consists of a sequence of four types of bases grouped into two disjoint pairs known as Watson-Crick complementary pairs that can bind together through hydrogen bonding in an operation known as hybridization. DNA enjoys a unique advantage for a nanostructure construction material because two single strands of DNA can be designed and constructed by the experimental scientist to be selectively sticky and bind together to form doubly stranded DNA. Hybridization is much more likely to occur if the DNA base sequences are comple-

mentarythat is, if the component bases are Watson-Crick pairs and the temperature and ionic composition of the solution are set appropriately. The resulting doubly stranded DNA is relatively rigid and forms the well-known double-helix geometry. If the sticky single-strand segments that hybridize abut doubly stranded segments of DNA, one can use an enzymic reaction known as ligation to concatenate these segments.

DNA Nanostructures. Seeman first pioneered DNA structure nanofabrication in the 1980s by assembling a multitude of DNA nanostructures (such as rings, cubes, and octahedrons) using DNA branched junctions and remains a leader in this area [38, 36, 39]. However, these early DNA nanostructures were not very rigid. To increase the rigidity of DNA nanostructures, Seeman made use of a DNA nanostructure known as a DNA crossover (also known as a *branched Holiday junction*), which consists of two doubly stranded DNA, each having a single strand that crosses over to the other. Pairs of crossovers, known as double crossovers, provide a significant increase in rigidity of a DNA nanostructure. Also, certain crossovers (known as antiparallel crossovers) cause a reversal in the direction of strand propagation following the exchange of the strand to a new helix.

DNA Tiles. These are quite rigid and stable DNA nanostructures that are formed from multiple DNA antiparallel crossovers. DNA tiles typically have a roughly rectangular geometry. These tiles come in multiple varieties that differ from one another in the geometry of strand exchange and the topology of the strand paths through the tile. The first DNA tiles developed were known as double-crossover(DX) tiles and composed of two DNA double helices with two crossovers [52]. LaBean, Reif, and Seeman have developed some novel DNA tiles known as triple-crossover (TX) tiles that are composed of three DNA double helices with four crossovers [15]. These TX tiles have properties that can facilitate one and two dimensional tiling assemblies and computations. Each DNA tile is designed to match the ends of certain other DNA tiles, a process that facilitates the assembly into tiling lattices. In particular, DNA tiles are designed to contain several short sections of unpaired, single-strand DNA (ssDNA) extending from the ends of selected helices (often called "sticky ends") that function as programmable binding domains, which are the *tile pads*. Both double- and triple-crossover tiles are useful for doing tiling assemblies. The DX tiles provide up to four pads for encoding associations with neighboring tiles, whereas the TX tiles provide up to six pads that are designed to function as binding domains with other DNA tiles. Use of pads with complementary base sequences provides control for the neighbor relations of tiles in the final assembly. In particular, the tile pads hybridize to the pads of other chosen DNA tiles. Individual tiles interact by binding with other specific tiles through hybridization of their pads to self-assemble into desired superstructures.

DNA Tiling Lattices. These are superstructures built up from smaller component structures (DNA tiles). Individual DNA tiles interact by annealing with other specific tiles via their ssDNA pads to self-assemble into desired superstructures. These lattices can be either: **(a)** *non-computational*, containing a fairly small number of distinct tile types in a repetitive, periodic pattern; or **(b)** *computational*, containing a larger number of tile types with more complicated association rules which perform a computation during lattice assembly. The direct assembly of DNA lattices from component single strand

DNA has been demonstrated for non-computational DNA lattices described below. Winfree and Seeman demonstrated the self-assembly of two-dimensional periodic lattices consisting of at hundreds of thousands of double-crossover tiles, which is strong evidence of this approach's scalability [52]. In addition, LaBean, Reif, and Seeman have constructed DNA TX molecules which produced tiling lattices of even larger numbers of tiles [15]. Both classes of self-assembled DNA lattices were observed through atomic force microscopy (AFM), a mechanical scanning process that provides images of molecular structures on a two-dimensional plate, as well as by use of transmission electron microscopy (TEM). Distinguishing surface features can be designed into individual tiles by slightly modifying the DNA strands comprising the tiles. These modified DNA strands form short loops that protrude above the tile. To enhance definition, we have also affixed metallic balls to these DNA loops using known methods for affixing gold balls to DNA. Surface features, such as two-dimensional banding patterns, have been programmed into these DNA lattices using DNA tiles that assemble into regular repetitive patterns. These topographical features were observed on the DNA tiling lattices with atomic force and transmission electron microscopy imaging devices [23, 20, 52]. These tiling assemblies had no fixed limit on their size. Recall that Reif introduced the concept of a *nano-frame*, which is a self-assembled nanostructure that constrains the subsequent timing assembly (*e.g.*, to a fixed size rectangle) [26]. A tiling assembly might be designed to be *self-delineating* (growing to only a fixed size) by the choice of tile pads that essentially "count" to their intended boundaries in the dimensions to be delineated. In addition, our lab recently developed a "waffle"-like DNA lattice composed of a novel type of DNA tiles (4 x 4 tile) [56]. We further used the 4 x 4 tiling lattices as templates for organizing nanoscale ligands, *e.g.* proteins and gold nano-particles [18, 25]. In addition, we have recently developed a new method for the assembly of aperiodic patterns [55].

Directed Nucleation Assembly Techniques. We have recently developed another method for assembly of complex patterns, where an input DNA strand is synthesized that encodes the required pattern, and then specified tiles assemble around blocks of this input DNA strand, forming the required 1D or 2D pattern of tiles [55]. This method uses artificially synthesized DNA strands that specify the pattern and around which 2D DNA tiles assemble into the specified pattern; in this method, the permanent features of the 2D pattern are generated uniquely for each case.

Computation by DNA Self-assembly. We now focus on another approach: computation by self-assembly. Adleman made use of a simple form of computation by self-assembly in his original experiment [1]: instead of blindly generating all possible sequences of vertices; instead, the oligonucleotide sequences and the logic of Watson-Crick complementarity guide the self-assembly processes so that only valid paths are generated.

Programming Self-assembly of DNA Tilings. Programming DNA self-assembly of tilings amounts to the design of the pads of the DNA tiles (recall these are sticky ends of single strand DNA that function as programmable binding domains, and that individual tiles interact by annealing with other specific tiles via their single strand DNA pads to self-assemble into desired superstructures). The use of pads with complemen-

tary base sequences allows the neighbor relations of tiles in the final assembly to be intimately controlled; thus the only large-scale superstructures formed during assembly are those that encode valid mappings of input to output. The self-assembly approach for computation only uses four laboratory steps:(i) mixing the input oligonucleotides to form the DNA tiles, (ii) allowing the tiles to self-assemble into superstructures, (iii) ligating strands that have been co-localized, and (iv) then performing a single separation to identify the correct output.

The Speed of Computing via DNA Tiling Assemblies (Compared with Silicon-based Computing). The speed of DNA tiling assemblies is limited by the annealing time, which can be many minutes, and can be 10^{10} slower than a conventional computer. A DNA computation via self-assembly must take into account the fact that the time to execute an assembly can range from a few minutes up to hours. Therefore, a reasonable assessment of the power of DNA computation must take into account both the speed of operation as well as the degree of massive parallelism. Nevertheless, the massive parallelism (both within assemblies and also via the parallel construction of distinct assemblies) possibly ranging up to 10^{18} provides a potential that may be advantageous for classes of computational problems that can be parallelized.

String-Tiles: A Mechanism for Small-Depth Tiling. An approach for small-depth computations is to compress several tile layers into single tiles, so that the simplest form of linear self-assembly suffices. Linear self-assembly schemes for integer addition were first described by [26]; in this scheme each tile performed essentially the operation of a single carry-bit logic step. This linear self-assembly approach works particularly well when the topology and routing of the strands in the DNA tiles is carefully considered, leading to the notion of string tiles. The concept of string tile assemblies derives from the observation that allowing neighboring tiles in an assembly to associate by two sticky ends on each side, one could increase the computational complexity of languages generated by linear self-assembly [51] showed that by allowing contiguous strings of DNA to trace through individual tiles and the entire assembly multiple times, surprisingly sophisticated calculations can be performed with one-layer linear assemblies of string tiles. The TAE tiles recently developed by LaBean [15] are particularly useful as string tiles. An experimental demonstration of the string tiles was achieved in our lab [54].

Input/Output to Tiling Assemblies Using Scaffold and Reporter Strands. Recall that the TX tiles are constructed of three double-helices linked by strand exchange. The TX tiles have an interesting property, namely that certain distinguished single stranded DNA (to be called scaffold and reporter strands, respectively) wind through all the tiles of a tiling assembly. This property provides a more sophisticated method for input and output of DNA computations in string tiling assemblies. In particular, there are two types. The TAE tile contains an Even (and the TAO tiles contains an Odd) number of helical half-turns between crossover points. Even spacing of crossovers of the TAE tile allows reporter strands to stretch straight through each helix from one side of the tile to the other. These reporter segments are used for building up a long strand which records inputs and outputs for the entire assembly computations.

(a) Input via Scaffold Strands: We take as input the scaffold strands and which encode the data input to the assembly computation. They are long DNA strands capable of serving as nucleation points for assembly. Preformed, multimetric scaffold strands are added to the hybridization/annealing mixture in place of the monomeric oligo corresponding to the tile's reporter segment. The remaining portion of the component ssDNA comprising the tiles are also added. In the resulting annealing process, tiles assemble around the scaffold strand, automatically forming a chain of connected tiles which can subsequently be used as the input layer in a computational assembly.

(b) Output via Reporter Strands: After ligation of the tiling assembly (this joins together each tile's segments of the reporter strands), the reporter strand provides an encoding of the output of the tiling assembly computation (and typically also the inputs). Note this input/output can occur in parallel for multiple distinct tiling assemblies. Finally, the tiling assembly is disassembled by denaturing (*e.g.*, via heating) and the resulting ssDNA Reporter Strands provide the result (these may be used as scaffold strands for later cycles of assembly computation, or the readout may be by PCR, restriction cutting, sequencing, or DNA expression chips).

One Dimensional DNA Tiling Computations for Parallel Arithmetic. We now outline procedures for using the string tiles described above that self-assemble into linear tiling assemblies to perform massively parallel arithmetic. LaBean *et. al.* describes tile systems that compute binary number addition (where the binary numbers are encoded by strands of DNA) by using two distinct sets of sticky-ends between adjacent tiles in the assembly to effectively communicate the values of the carry-bits [14]. (They can also be used for computation of bit-wise XOR of Boolean vectors encoded by strands of DNA.) The assemblies result in the appending of these strands to the addition sums. For computations on specific inputs, these procedures make use of the scaffold strands mentioned above. The inputs are self-assembled strands of DNA composed of sequences DNA words encoding the pairs of binary numbers to be summed. Otherwise, the input tiles can be (using known techniques uses for the assembly of combinatorial libraries of DNA strands) randomly assembled and thereby generate a molecular look-up table in which each reporter strand encodes the random inputs and resultant outputs of a single calculation. After denaturing the assemblies back to individual strands, one may sample the resulting reporter strands to verify the outputs are correctly computed. A sufficient number of DNA tile molecules provide full coverage of all possible n-bit input strings. Such look-up tables may be useful as input for further computations as they represent a unique library of sequences with a complex structural theme. An experimental demonstration of an XOR tiling computation based on TAO tiles is reported in [22].

Two Dimensional DNA Tiling Computations. In the immediate future, it may be possible to extend the one dimensional DNA tiling assembly methods to two dimensional tilings, and to demonstrate these methods experimentally. One interesting goal is integer multiplication. The most direct and relatively straightforward way is to multiply via repeated additions and bit shifts, applying known VLSI systolic array architecture designs for integer multiplication. This would require a two dimensional $n \times n$ tiling assembly, with some increased complexity over the linear assembly for integer addition. On the other hand, it will provide the first demonstration of computation of a two dimen-

sional DNA self-assembly. Two dimensional computational tilings may also be used to do logical processing. Lagoudakis and LaBean proposed a 2D DNA self-assembly for Boolean variable satisfiability, which uses parallel construction of multiple self-assembling 2D DNA lattices to solve the problem [16]. Such methods for solving combinatorial search problems do not scale well with the input size (the number of parallel tiling assemblies grows exponentially with the number of Boolean variables of the formula). However, similar constructions may be used for evaluating Boolean queries and circuits in massively parallel fashion, for multiple input settings of the input Boolean variable, and in this context it may be appropriate to consider the Boolean formula to be of fixed size.

Three Dimensional DNA Tiling Computations. There is a number of possible methods for executing computations experimentally on 3D DNA lattices, providing computations with (implicit) data movement in three dimensions. Matrix inner product might be executed by a three dimensional computational tiling by applying known VLSI systolic array architecture designs for matrix inner product. Another possible three dimensional computational tiling is that of the time-evolution (time is the third dimension of the tiling) of a 2D cellular automata, *e.g.*, 2D cellular automata simulation of fluid flow.

DNA Robotics. Existing DNA nanomechanical devices can exhibit motions such as open/close [42, 43, 63], extension/contraction [5, 9, 19], and rotation [24, 57]. These motions are mediated by external environmental changes such as the addition and removal of DNA fuel strands [5, 9, 19, 42, 43, 57, 63] or the change of ionic strength of the solution [24]. Our lab has recently constructed a robust sequence-dependent DNA nanomechanical actuator and have incorporated it into a 2D parallelogram DNA lattice [23]. The actuator can be switched reversibly between two states, mediated by the addition and removal of fuel DNA strands.

An improvement of the above devices is the construction of DNA nanomechanical devices that achieve autonomous and non-localized motions, *e.g.* walking motion. Turberfield and colleagues have designed a free running DNA machine [44] using DNA as fuels. Mao's group recently constructed autonomous DNA tweezers powered by a DNA enzyme [8]. Seeman's group and Pierce's group respectively constructed a non-autonomous DNA walking device powered by the addition and removal of DNA fuel strands [40, 41]. In our group, Reif designed an autonomous DNA walking device and an autonomous DNA rolling device that move in a random bidirectional fashion along DNA tracks [28]. Building on Reif's original designs, we designed a suite of unidirectional autonomous DNA walking devices [60] and experimentally implemented one walking device in which a DNA fragment makes a 3-stop unidirectional motion along a self-assembled linear DNA track autonomously [62]. Based on this device, we further designed an autonomous universal DNA Turing machine [61] and autonomous universal DNA cellular automata [59]; their operations were verified with computer simulation.

6   Conclusion

The self-assembly of DNA is a promising emerging method for molecular scale constructions and computations. We have overviewed the area of DNA tiling self-

assemblies and noted a number of open problems. We have discussed the potential approaches for error-control in self-assembly techniques for DNA computation; particularly the use of error-resilient modified tiling methods. We have identified some technological impacts of DNA assemblies, such as using them as platform for constructing molecular electronic and robotic devices. Important future work includes further investigating potential broader technological impacts of DNA lattices. Many applications of DNA lattices rely on the development of appropriate attachment methods between DNA lattice and other nanoparticles, which itself is a key challenge in DNA based nanoscience.

Acknowledgement

References

1. L. Adleman. Molecular computation of solutions to combinatorial problems. *Science*, 266:1021–1024, 1994.
2. L. Adleman. Towards a mathematical theory of self-assembly. Technical Report 00-722, University of Southern California, 2000.
3. L. Adleman, Q. Cheng, A. Goel, M.D. Huang, D. Kempe, P.M. de Espans, and P.W.K. Rothemund. Combinatorial optimization problems in self-assembly. In *Proceedings of the thirty-fourth annual ACM symposium on Theory of computing*, pages 23–32. ACM Press, 2002.
4. L. Adleman, Q. Cheng, A. Goel, M.D. Huang, and H. Wasserman. Linear self-assemblies: Equilibria, entropy, and convergence rate. In *Sixth International Conference on Difference Equations and Applications*, 2001.
5. P. Alberti and J.L. Mergny. DNA duplex-quadruplex exchange as the basis for a nanomolecular machine. *Proc. Natl. Acad. Sci. USA*, 100:1569–1573, 2003.
6. Y. Benenson, T. Paz-Elizur, R. Adar, E. Keinan, Z. Livneh, and E. Shapiro. Programmable and autonomous computing machine made of biomolecules. *Nature*, 414:430–434, 2001.
7. R. Berger. The undecidability of the domino problem. *Memoirs of the American Mathematical Society*, 66, 1966.
8. Y. Chen, M. Wang, and C. Mao. An autonomous DNA nanomotor powered by a DNA enzyme. *Angew. Chem. Int. Ed.*, 43:3554–3557, 2004.
9. L. Feng, S.H. Park, J.H. Reif, and H. Yan. A two-state DNA lattice switched by DNA nanoactuator. *Angew. Chem. Int. Ed.*, 42:4342–4346, 2003.
10. B. Grunbaum and G.C. Shepard. *Tilings and Patterns*, chapter 11. H Freeman and Company, 1986.
11. N. Jonoska and S. Karl. Ligation experiments in computing with dna. In *Proceedings of 1997 IEEE International Conference on Evolutionary Computation (ICEC'97)*, pages 261–265, 1997.
12. N. Jonoska, S.A. Karl, and M. Saito. Graph structures in dna computing. *Computing with Bio-Molecules, theory and experiments*, pages 93–110, 1998.

13. T.H. LaBean. Introduction to self-assembling DNA nanostructures for computation and nanofabrication. In *Computational Biology and Genome Informatics eds. J.T.L. Wang and C.H. Wu and and P. P. Wang ISBN 981-238-257-7 World Scientific Publishing Singapore*, 2003.

14. T.H. LaBean, E. Winfree, and J.H. Reif. Experimental progress in computation with DNA molecules. In *Proc. DNA Based Computers V*, 1999.

15. T.H. LaBean, H. Yan, J. Kopatsch, F. Liu, E. Winfree, J.H. Reif, and N.C. Seeman. The construction, analysis, ligation and self-assembly of DNA triple crossover complexes. *J. Am. Chem. Soc.*, 122:1848–1860, 2000.

16. M.G. Lagoudakis and T.H. LaBean. 2-D DNA self-assembly for satisfiability. In *DNA Based Computers V*, volume 54 of *DIMACS*, pages 141–154. American Mathematical Society, 2000.

17. H.R. Lewis and C.H. Papadimitriou. *Elements of the Theory of Computation*. Prentice-Hall, 1981.

18. H. Li, S.H. Park, J.H. Reif, T.H. LaBean, and H. Yan. DNA templated self-assembly of protein and nanoparticle linear arrays. *Journal of American Chemistry Society*, 126(2):418–419, 2004.

19. J. Li and W. Tan. A single DNA molecule nanomotor. *Nano Lett.*, 2:315–318, 2002.

20. D. Liu, S.H. Park, J.H. Reif, and T.H. LaBean. DNA nanotubes self-assembled from triple-crossover tiles as templates for conductive nanowires. *Proceedings of the National Academy of Science*, 101:717–722, 2004.

21. Q. Liu, L. Wang, A.G. Frutos, A.E. Condon, R.M. Corn, and L.M. Smith. DNA computing on surfaces. *Nature*, 403:175–179, 2000.

22. C. Mao, T.H. LaBean, J.H. Reif, and N.C. Seeman. Logical computation using algorithmic self-assembly of DNA triple-crossover molecules. *Nature*, 407:493–496, 2000.

23. C. Mao, W. Sun, and N.C. Seeman. Designed two-dimensional DNA holliday junction arrays visualized by atomic force microscopy. *J. Am. Chem. Soc.*, 121:5437–5443, 1999.

24. C. Mao, W. Sun, Z. Shen, and N.C. Seeman. A DNA nanomechanical device based on the B-Z transition. *Nature*, 397:144–146, 1999.

25. S.H. Park, P. Yin, Y. Liu, J.H. Reif, T.H. LaBean, and H. Yan. Programmable DNA self-assemblies for nanoscale organization of ligands and proteins. 2004. Submitted for publication.

26. J.H. Reif. Paradigms for biomolecular computation. In C. S. Calude, J. Casti, and M. J. Dinneen, editors, *First International Conference on Unconventional Models of Computation, Auckland, New Zealand*, pages 72–93. Springer Verlag, 1998.

27. J.H. Reif. Local parallel biomolecular computation. In H. Rubin and D. H. Wood, editors, *DNA-Based Computers 3*, volume 48 of *DIMACS*, pages 217–254. American Mathematical Society, 1999.

28. J.H. Reif. The design of autonomous DNA nanomechanical devices: Walking and rolling DNA. *Lecture Notes in Computer Science*, 2568:22–37, 2003. Published in Natural Computing, DNA8 special issue, Vol. 2, p 439-461, (2003).

29. J.H. Reif, S. Sahu, and P. Yin. Compact error-resilient computational DNA tiling assemblies. *Tenth International Meeting on DNA Based Computers (DNA10)*, 2004.

30. J.H. Reif. The emergence of the discipline of biomolecular computation. *Biomolecular Computing, New Generation Computing*, 20(3):217–236, 2002.

31. J.H. Reif. Molecular assembly and computation: From theory to experimental demonstrations. In *29-th International Colloquium on Automata, Languages, and Programming(ICALP), Mlaga, Spain*, pages 1–21, 2002.

32. J.H. Reif, T.H. LaBean, and N.C. Seeman. Challenges and applications for self-assembled dna nanostructures. In *Lecture Notes in Computer Science*, volume 2054, pages 173–198. Springer-Verlag, Berlin Heidelberg, 2001.

33. R.M. Robinson. Undecidability and non periodicity of tilings of the plane. *Inventiones Math*, 12:177–209, 1971.

34. P.W.K. Rothemund and E. Winfree. The program-size complexity of self-assembled squares (extended abstract). In *Proceedings of the thirty-second annual ACM symposium on Theory of computing*, pages 459–468. ACM Press, 2000.

35. N.C. Seeman. *De novo* design of sequences for nucleic acid structural engineering. *J. Biomol. Struct. Dyn.*, 8:573–581, 1990.

36. N.C. Seeman. Nucleic acid nanostructures and topology. *Angew. Chem. Int. Ed.*, 37:3220–3238, 1998.

37. N.C. Seeman. DNA in a material world. *Nature*, 421:427–431, 2003.

38. N.C. Seeman, Y. Zhang, and J. Chen. DNA nanoconstructions. *J. Vac. Sci. Technol.*, 12:4:1895–1903, 1994.

39. R. Sha, F. Liu, M.F. Bruist, and N.C. Seeman. Parallel helical domains in DNA branched junctions containing 5', 5' and 3', 3' linkages. *Biochemistry*, 38:2832–2841, 1999.

40. W.B. Sherman and N.C. Seeman. A precisely controlled DNA biped walking device. *Nano Lett.*, 4:1203–1207, 2004.

41. J.S. Shin and N.A. Pierce. A synthetic DNA walker for molecular transport. *J. Am. Chem. Soc.*, 126:10834–10835.

42. F.C. Simmel and B. Yurke. Using DNA to construct and power a nanoactuator. *Phys. Rev. E*, 63:041913, 2001.

43. F.C. Simmel and B. Yurke. A DNA-based molecular device switchable between three distinct mechanical states. *Appl. Phys. Lett.*, 80:883–885, 2002.

44. A.J. Turberfield, J.C. Mitchell, B. Yurke, Jr. A.P. Mills, M.I. Blakey, and F.C. Simmel. DNA fuel for free-running nanomachines. *Phys. Rev. Lett.*, 90:118102, 2003.

45. H. Wang. Proving theorems by pattern recognition ii. *Bell Systems Technical Journal*, 40:1–41, 1961.

46. E. Winfree. Complexity of restricted and unrestricted models of molecular computation. In R. J. Lipton and E. B. Baum, editors, *DNA Based Computers*, volume 27 of *DIMACS*, pages 187–198. American Mathematical Society, 1995.

47. E. Winfree. On the computational power of DNA annealing and ligation. In R. J. Lipton and E. B. Baum, editors, *DNA Based Computers 1*, volume 27 of *DIMACS*, pages 199–221. American Mathematical Society, 1996.

48. E. Winfree. *Algorithmic Self-Assembly of DNA*. PhD thesis, California Institute of Technology, 1998.

49. E. Winfree. Simulation of computing by self-assembly. Technical Report 1998.22, Caltech, 1998.

50. E. Winfree and R. Bekbolatov. Proofreading tile sets: logical error correction for algorithmic self-assembly. In *DNA Based Computers 9*, volume 2943 of *LNCS*, pages 126–144, 2004.

51. E. Winfree, T. Eng, and G. Rozenberg. String tile models for DNA computing by self-assembly. In *DNA Based Computers 6*, pages 63–88, 2000.

52. E. Winfree, F. Liu, L.A. Wenzler, and N.C. Seeman. Design and self-assembly of two-dimensional DNA crystals. *Nature*, 394(6693):539–544, 1998.

53. E. Winfree, X. Yang, and N.C. Seeman. Universal computation via self-assembly of DNA: Some theory and experiments. In L. F. Landweber and E. B. Baum, editors, *DNA Based Computers II*, volume 44 of *DIMACS*, pages 191–213. American Mathematical Society, 1996.

54. H. Yan, L. Feng, T.H. LaBean, and J.H. Reif. Parallel molecular computation of pair-wise xor using DNA string tile. *J. Am. Chem. Soc.*, 125(47), 2003.

55. H. Yan, T.H. LaBean, L. Feng, and J.H. Reif. Directed nucleation assembly of DNA tile complexes for barcode patterned DNA lattices. *Proc. Natl. Acad. Sci. USA*, 100(14):8103–8108, 2003.

56. H. Yan, S.H. Park, G. Finkelstein, J.H. Reif, and T.H. LaBean. DNA-templated self-assembly of protein arrays and highly conductive nanowires. *Science*, 301(5641):1882–1884, 2003.

57. H. Yan, X. Zhang, Z. Shen, and N.C. Seeman. A robust DNA mechanical device controlled by hybridization topology. *Nature*, 415:62–65, 2002.

58. P. Yin, B. Guo, C. Belmore, W. Palmeri, E. Winfree, T.H. LaBean, and J.H. Reif. TileSoft: Sequence optimization software for designing DNA secondary structures. Technical Report CS-2004-09, Duke University, Computer Science Department, 2004.

59. P. Yin, S. Sahu, A.J. Turberfield, and J.H. Reif. Design of autonomous DNA cellular automata. 2004. In preparation.

60. P. Yin, A.J. Turberfield, and J.H. Reif. Designs of autonomous unidirectional walking DNA devices. In *DNA Based Computers 10*, 2004.

61. P. Yin, A.J. Turberfield, S. Sahu, and J.H. Reif. Design of an autonomous DNA nanomechanical device capable of universal computation and universal translational motion. In *DNA Based Computers 10*, 2004.

62. P. Yin, H. Yan, X.G. Daniell, A.J. Turberfield, and J.H. Reif. A unidirectional DNA walker moving autonomously along a linear track. *Angew. Chem. Int. Ed.*, 43:4906–4911, 2004.

63. B. Yurke, A.J. Turberfield, Jr. A.P. Mills, F.C. Simmel, and J.L. Neumann. A DNA-fuelled molecular machine made of DNA. *Nature*, 406:605–608, 2000.