Paradigms for Biomolecular Computation

John H. Reif
Department of Computer Science
Duke University *

Abstract

Biomolecular Computation (BMC) is computation done at the molecular scale, using Biotechnological techniques. This paper discusses the underlying biotechnology that BMC may utilize, and surveys a number of distinct paradigms for doing BMC. We also identify a number of key future experimental milestones for the field of BMC.

1 Introduction

BMC is a new field, with largely unexplored methodologies. It is inter-disciplinary by nature, lying in the interface between biochemistry and computer science. We discuss a number of distinct methods for BMC. All these methods use Biotechnological techniques to do computation or processing at the molecular scale. We will mostly discuss DNA methods for BMC (this is also termed *DNA computation*). See Gifford [G94], Smith and Schweitzer [SS95], Rubin [R96] and Delcher, Hood, and Karp [DHK96] for previous surveys and summaries of the field of BMC.

Goals and Potential Applications of BMC. Here we discuss problems (up to moderate sizes) that may benefit from the massive parallelism and nano-scale miniaturization available to BMC.

• **NP search problems.** These are a class of computational problems apparently requiring a large combinatorial search for their solution, but requiring modest work to verify a correct solution. NP search problems may be solved by BMC by (i) assembling a large number of potential solutions to the search problem, where each potential solution is encoded on a distinct strand of DNA, and (ii) then performing recombinant DNA operations which separate out the correct solutions of the problem. Adleman [A94] was the first to make use of BMC to solve a computational problem, in particular the Hamiltonian graph problem. He experimentally verified his method on a 7 node instance of this NP search problem. This was the first major experimental milestone achieved in the field of BMC.

• **Huge Memories.** BMC has the potential to provide huge memories. Each individual strand of DNA can encode binary information. A small volume can contain a vast number of molecules. As we shall discuss in Section 2, DNA in weak solution in one liter of water can encode 10^7 to 10^8 tera-bytes, and we can perform massively parallel associative searches (see Baum [B95] and Reif and LaBean [RL00]) on these memories.

*Surface address: Department of Computer Science, Duke University, Box 90129, Durham, NC 27708-0129. E-mail: reif@cs.duke.edu. This work was supported by Grants NSF/DARPA CCR-9725021, CCR-96-33567, NSF IRI- 9619647 and EIA-0086015, ARO contract DAAH-04-96-1-0448, and ONR contract N00014-99-1-0406. This paper appears in the First International Conference on Unconventional Models of Computation, Auckland, New Zealand, January 1998. Proceedings published in Unconventional Models of Computation, edited by C.S. Calude, J. Casti, and M.J. Dinneen, Springer Publishers, 1998, pp 72-93. Postscript version: http://www.cs.duke.edu/~reif/paper/paradigm.ps

• **Massively Parallel Machines.** BMC also has the potential to supply massive computational power. General use of BMC is to construct parallel machines where each processor's state is encoded by a DNA strand. BMC can perform massively parallel computations by executing recombinant DNA operations that act on all the DNA molecules at the same time. These recombinant DNA operations may be performed to execute massively parallel local memory read/write, logical operations and also further basic operations on words such as parallel arithmetic. As we discuss in Section 2, DNA in weak solution in one liter of water can encode the state of about 10^{18} processors, and since certain recombinant DNA operations can take many minutes, the overall potential for a massively parallel BMC machines is about $1,000$ tera-ops. (This assumes the parallel machine uses local rather than global shared memory. To allow such a parallel machine to use global shared memory, we need to do massively parallel message (DNA strand) routing. Reif's [R95] BMC simulation of a PRAM with shared memory required volume growing at least quadratically with size of the storage of the PRAM, but Gehani and Reif [GR98a] describe a MEMS micro-flow device technology that can do the massively parallel message routing with a substantial decrease in the volume.)
• **DNA Nano-fabrication and Self-assembly.** BMC techniques combined with DNA nano-fabrication techniques may allow for the self-assembly of DNA tiles into lattices in 2 and 3 dimensions and the construction of complex nano-structures that encode computations (see Reif et al [RLS00]).
• **Processing of Natural DNA.** BMC techniques may also be used in problems that are not implicitly digital in nature, for example the processing of natural (biologically derived) DNA. These techniques may be used to provide improved methods for the sequencing and fingerprinting of natural DNA, and the solution of other biomedical problems. The results of processing natural DNA can be used to form *wet data bases* with re-coded DNA in solution, and BMC can be used to do fast searches and data base operations on these wet databases (see Baum [B95] and Reif and LaBean [RL00]).
Organization. In Section 1, we briefly discuss the general goals and potential applications of BMC. In Section 2, we discuss biotechnology for BMC, including ways in which conventional biotechnology may need to be tailored for BMC. Then in successive sections we discuss various distinct paradigms for BMC. We consider in Section 3 the splicing paradigm for BMC, which provides a theoretical model of enzymatic systems operating on DNA, was the first paradym for BMC to be proposed, then discuss in Section 4 the distributed molecular parallism paradigm for BMC, and in Section 5 discuss the local assembly paradigm for BMC, which does computation by assembly of DNA tiles. In Section 6 we discuss a cellular processing paradigm where BMC is done using a microorganism such as bacteria to do computation, by re-engineering the regulatory feedback systems used in cellular metabolism. Finally, in Section 7 we discuss various biological applications of BMC and the DNA^{2}DNA paradigm. In Section 8 we conclude the paper with a comparison of the current state of BMC with the state of the field of VLSI in the 1970's, and with mention of some other alternative paradigms for BMC.

2 Biotechnology for BMC.

DNA Hybridization DNA is a molecule consisting of a linear sequence of nucleotides. There are 4 types of nucleotides, which are complementary in pairs. A key property of DNA is *Watson-Crick complementation*, which allows the binding of complementary nucleotides. DNA may be single stranded (ssDNA) or double stranded. An ssDNA has an orientation $3' - 8'$ or $5' - 3'$. If two ssDNA are Watson-Crick complementary and $3' - 8'$ and $5' - 3'$ oriented in opposite directions, they are said to be *sticky*. At the appropriate conditions (determined by temperature and salinity, etc.), they may hybridize into double-stranded DNA. This resulting double-stranded DNA has complementary strands in opposite orientation. This allows the *annealing* of large strands of single DNA into double DNA, and the formation of complex 3D structures (this is known as *secondary structure*). The reverse process (usually induced by heating) is the *denature* of complex structures into single stranded linear structures. See [MH87, PP97, W97, RDGS97, HG97] for mathematical models of DNA hybridization and their simulation via thermodynamics.

Recombinant DNA Technology. In the last two decades, there have been revolutionary advances in the field of biotechnology. Biotechnology has developed a large set of procedures for modifying DNA, known collectively known as *recombinant DNA*.

Short strands of ssDNA of length n are sometimes called *n-mers*. Many recombinant DNA operations use hybridization and are specific to a DNA segment with a prescribed n-mer subsequence. Such recombinant DNA operations include *cleavage* of DNA strands, *separation* of DNA strands, *detection* of DNA strands, and *fluorescent tagging* of specific DNA words. In addition, there are operations that are not specific, including *ligation* of DNA segments to form covalent bonds that join the DNA strands together, *merging* of test tube contents, the denature operation discussed above, and separation by molecular weight. Basic principles of recombinant DNA technology are described in [WGWZ92] [WHR87, OP94]. Detailed theoretical discussions of dynamics, thermodynamics, and structure of DNA, RNA and certain proteins are given by [BKP90, S94, EC]. Also see [ER82, MH87] for the dynamics and chemistry of nucleic acids and enzymes.

Due to the industrialization of the biotechnology field, laboratory techniques for recombinant DNA and RNA manipulation are becoming highly standardized, with well written lab manuals (e.g. [SFM89]) detailing the elementary lab steps required for recombinant DNA operations. Many of those recombinant DNA operations which where once considered highly sophisticated are now routine, and many have been automated by robotic systems. As a further byproduct of the industrialization of the biotechnology field, many of the constraints (such as timing, pH, and solution concentration, contamination etc.) critical to the successful execution of recombinant DNA techniques for conventional biological and medical applications (but not necessarily for all BMC applications), are now quite well understood, both theoretically and in practice.

Alternative Recombinant DNA Methodologies. An *enabling technology* is a technology that allows a task to be done. The most pervasive enabling biotechnology for BMC is solution-based recombinant DNA, that is the recombinant DNA operations are done on test tubes with DNA in solution. However, their are a number of alternative enabling biotechnologies, that allow similar and sometimes enhanced capabilities.

– **Solid Support BMC.** An example of an alternative recombinant DNA methodology is the *solid support* of individual DNA, for example by *surface attachments*. In solid support, the DNA strands are affixed to supports of some sort. In surface-based chemistry, surface attachments are used to affix DNA strands to organic compounds on the surface of a container. This can allow for more control of recombinant DNA operations, since this insures (i) that distinct DNA strands so immobilized can not interact, and also (ii) allows reagents and complementary DNA to have easy access to the DNA, and (iii) allows for easy removal of reagents and secondary by-products. Also, handling of samples is simpler and more readily automated. Surface-based chemistry has been used in protein sequencing, DNA synthesis, and peptide synthesis [S88]. Surface attachment methods can also be used for optical read-out (e.g., via fluorescent tagging of specific DNA words) on 2D arrays. A possible drawback of surface attachment technology, in comparison to solution-based recombinant DNA techniques, is a reduction on the total number of DNA strands that can be used.

– **Automation and Miniaturization of BMC.** MEMS is the technology of miniature actuators, valves, pumps, sensors and other such mechanisms, and when controlling fluids it is known as *MEMS micro-flow device technology.* [EE92, VSJMWR92, MEBH92]. Some of the current limits of BMC stem from the labor intensive nature of the laboratory work, the error rates, and the large volumes needed for certain bio-molecular reactions to occur (e.g., for searching and associative matching in wet data bases). [GR98a] (also see Ikuta [Iku96], Suyama [Suy98] for use of micro-flow devices for various biological applications) propose the use of MEMS micro-flow device technology for BMC which may provide several advantages: it would allow automation of the laboratory work, parallel execution of the steps of a BMC algorithm (for improved speed and reliability), and for transport of fluids and DNA among multiple micro-test tubes. [GR98a] provide a model for micro-flow based bio-molecular computation (MF-BMC) which uses abstractions of both the recombinant DNA (RDNA) technology as well as of the micro-flow technology, and takes into account both of their limitations (e.g., concentration limitations for reactants in RDNA, and the geometric limitations of the MEMS device fabrication technology). [GR98a] also give a time and volume efficient MF-BMC architecture for routing DNA strands among multiple micro-test tubes (this gives a substantial decrease in the volume required for the PRAM simulation of [R95]).

Physical Constraints for BMC using Recombinant DNA. The energy consumption, processing rate, and volume, are all important resources to consider in miniaturized and mobile computing devices, and in particular molecular scale computations. Conventional electronic supercomputers computers of the size of a work station operate in the range of 10^{-9} Joules per operation, at up to about 50 giga-ops per second, with memory of about 10 to 100 giga-bytes, and in a volume of about 10 cm^2.

What are the are the volume, time, and energy constraints for BMC, using recombinant DNA ?

• **Volume.** A small volume can contain a vast number of DNA molecules. A reasonable concentration is 5 grams of DNA per one liter of water. Then a liter of water contains in solution about 10^{21} DNA bases. For an associative memory (see Baum [B95] and Reif and LaBean [RL00]) of this scale, we can provide a few bytes of memory per DNA strand, by use of at most 100 base pairs per DNA strand. Thus a liter of solution provides an associative memory with 10^{19} to 10^{20} bytes, which is 10^7 to 10^8 tera-bytes. A DNA strand may need $1,000$ base pairs to encode a processor state and so a liter of solution encodes the state of approximately 10^{18} distinct processors.

• **Time.** The time duration of the recombinant DNA operations such as annealing, which depends on hybridization, is strongly dependent on the length of sequences to be matched and may also depend on temperature, pH, solution concentration, and possibly other parameters. These recombinant DNA and other Biotechnological operations can take up to 100 minutes.

• **Processing Speed.** Thus the overall potential for BMC using DNA is 10^{15} to 10^{16} operations per second in the liter of solution, which is $1,000$ tera-ops.

• **Energy.** Many recombinant DNA operations such as denaturing and annealing, are reversible, so they require arbitrarily small energy (they require heating or cooling, but this can be done using heat baths). Other recombinant DNA operations such as separation, do not seem to be reversible, and use approximately 10^{-19} Joules per operation.

Enhancement of Recombinant DNA for BMC. BMC has certain requirements not met by conventional recombinant DNA technology. Various methods have been developed which improve conventional recombinant DNA to obtain high yields and to allow for repeatability of operations. Also, analytic and simulation models of key recombinant DNA operations are being developed.

• **Efficient Error-resistant Separations.** Separation operations involve the isolation of all DNA with particular n-mer subsequences. Certain BMC methods require separation operations with high efficiency and high specificity. Approaches to solve this problem include the use of solid support, and most importantly the careful design of the n-mers used in separations. See Chen and Wood [CW97] and Deputat, Hajduczok, and Schmitt [KG97] for DNA separation techniques which may provide low error rates. Also see Boneh and Lipton [BL95a], Amos, Gibbons, and Hodgson [AGH96] and Deputat, Hajduczok, and Schmitt [DHS97] for methods that make BMC error resistant.

• **Ligation Errors.** Yoshinobu, et al [YAT+98] describe models for ligation errors and propose methods for compensating for them in BMC.

• **Word Design for BMC** is the problem of designing of a library of short n-mer sequences (DNA words) for information storage. Word design is crucial to error control in BMC. Ideally, a good word design will minimize unwanted secondary structure, and minimize mismatching, by maximizing binding specificity. Note that there are conflicting requirements on word design for BMC: as strand length decreases (which is desirable), the Hamming distance between distinct words of information decreases (which is not desirable). Adleman [A94] and Lipton [L94] first suggested the use of random strings for word design, noting that DNA strings are non-degenerate with high likelihood. Evolutionary search methods for word designs are described in [DMRGF+97]. Other word designs for BMC are described in [B96, DMGFS96, M96, GDNMF97]. Laboratory experiments of word designs are described in Libchaber [KCL96] and ligation experiments are described by Jonoska and Karl [JK97a]. Related issues in DNA computer system design have been addressed in [A96] by Amenyo. Word designs for surface-based chemistry is considered in [GFBCL+96] and in [FTCSC97], which provides a four-base mismatch word design. [CRFCC+96] shows that surface morphology may be an important factor for discrimination of mismatched DNA sequences. Wood [Woo98] considers the use of error correcting codes for word design and to decrease errors in BMC. Hartemink et al [HGL98] describes an automated constraint-based procedure for nucleotide sequence selection for BMC.

3 The Splicing Paradigm

Splicing is a paradigm for BMC which provides a theoretical model of enzymatic systems operating on DNA. Splicing models a solution with enzymatic actions (restrictions and the ligations) operating on DNA in parallel in a common test tube. The DNA strands are modeled as strings over a finite alphabet. Splicing allows for the generation of new strings from an initially chosen set of strings, using a specified set of splicing operations. The splicing operations on the DNA are string editing operations such as cutting, appending, etc. These operations can be applied in any order, and thus the splicing system can be considered to be autonomously controlled. Also, the operations may be nondeterministic, and a large number of possible results may be obtained from a small number of operations. Splicing predates all other BMC paradigms and it has its roots in formal language theory [H87] and Lindenmayer systems [H92]. Pioneering work on splicing was done by Head [H92]. There is now a rather extensive literature (including thirty or so papers) in splicing and related string rewrite systems, written by a dozen researchers, including Paun [P96a, P96b, P97] and Paun, et al [CFKP96, HPP96, PRS96], Culik and Harju [CH89] Pixton[Pi95, Pi96, Pi97], [StM97], Yokomori and Kobayashi [YK97a, YK97b], Kim and Kyungpook [KK97]. All of these investigations were theoretical in nature, and established the computational power of splicing systems of various sorts. For example, [HPP96] provided solution of the characterization problem for splicing system H(Fin,Fin). A number of researchers, including Csuhaj-Varju, Freund, Kari, and Paun [CFKP96, Kar97A, FKP98], Rothemund [Ro95], and Smith and Schweitzer [SS95] independently proved that a universal Turing Machine can be simulated by recombinant DNA operations in splicing models. Also, Kari, Paun, Rozenberg, Salomaa, Yu proved that DNA sticker systems are universal [KPRS98]. Garzon and Jonoska, [GJ98] (also see Fu, Beigel [FB98]) characterize the complexity of splicing with strands of bounded length, to be PSPACE. Manca et al [ADL+98] give some further splicing models and Conrad [Con98] considers context free and context sensitive splicing methods. Margenstern and Rogozhin [MR98] consider time-varying splicing systems. Li [Li98]gives an algebraic characterization of certain splicing languages. Landweber and Kari [LK98] present a splicing model for the natural DNA editing and compression that occurs in certain protozoa. Surveys of DNA computing in the context of the splicing model are given by Kari

[Kar98, Kar97B, Kar96] and Kari, Sakakibara [KarS97].

In summary, splicing provided the first theoretical studies of BMC and and has contributed to our understanding of the potential power of BMC. It has evolved to be a very active subfield of formal language theory. At this time, splicing is primarily a theoretical rather than an experimental area of BMC. There are a number of practical issues (e.g., the number of distinct enzymes with distinct recognition sequences for DNA splicing operations are limited to at most a few hundred) that may limit the scale of experimental implementations of splicing, but it is quite possible that evolutionary techniques (using RNA enzymes) may be used to solve such difficulties. Recently an experimental test of splicing was done by Laun and Reddy [LR97], which provided a laboratory demonstration of splicing, testing a system with enzymatic actions (restrictions and the ligations) operating on DNA in parallel in a test tube.

4 The Distributed Molecular Parallelism Paradigm

In the *distributed molecular parallelism (DP-BMC)* paradigm for BMC, the operations are executed in parallel on a large number of distinct molecules in a distributed manner, using the massive parallelism inherent in BMC. Feynman [F 61] first proposed doing computation via distributed molecular parallelism, but his idea was not tested for a number of decades.

Solving NP Search Problems using DP-BMC. Adleman was the first to actually do an experiment demonstrating BMC, solving a small NP search problem. The *Hamiltonian path* problem is to find a path in a graph that visits each node exactly once. Adleman [A94] (also see [G94, KTL97] and Fu et al [FBZ98] for improvements to Adleman's [A94] Hamiltonian path BMC experiment, and see [MoS97] for related methods) employed molecular parallelism in the solution of the Hamiltonian path problem, by forming a set of DNA strings encoding sequences of nodes of the graph and restricting the set of sequences to only Hamiltonian paths. The number of recombinant DNA operations grew linearly with the size of the input graph. In a lab experiment, he tested his techniques on DNA for a 7 node Hamiltonian path problem. As previously stated, this was the first major experimental milestone achieved in the field of BMC.

Many subsequent methods for BMC have also made use of the DP-BMC paradigm. The *SAT problem* is to find variable assignments that satisfy a Boolean formula. Lipton [L94] proposed use of DP-BMC for finding satisfying inputs to a Boolean expression, and this approach was generalized in [BDLS95] to solve the SAT problem (also Eng [Eng98] proposed in vivo BMC methods for SAT). Jonoska and Karl [JK96] use DP-BMC to solve graph coloring problems. Gloor et al [GKG+98] give a BMC combinatorial search algorithm for the NP complete shortest common superstring problem. Beaver [Be94] proposed similar use of DP-BMC in the solution of the integer factorization problem. Both [BDL95] and [ARRW96] propose DP-BMC methods for breaking the DES cryptosystem. Conrad and Zauner [CZ97] propose DP-BMC methods for protein conformation.

• **Surface-Based NP search.** Eng, and Serridge [ES97] give a surface-based DP-BMC algorithm for minimal set cover. Wang [WQF+98] describe the experimental execution, within surface based BMC, of the operations: DESTROY and READOUT DNA computing operations: DESTROY and READOUT using a one word approach to solve a satisfiability problem. Liu et al [LFW+98] give an experimental demonstration of surface based BMC using a one word approach to solve a SAT problem.

• **NP search using RNA.** Recently Cukras, Faulhammer, Lipton, and Landweber [CFL+98] gave an impressive experimental demonstration of a BMC method for the solution of a class of SAT problems (derived from the knights problem in Chess), that appears likely to scale to at least moderate number of Boolean variables (say 18 to 24). Their method was also significant due to their use of RNA rather than DNA and their development of a powerful evolutionary method for doing the combinatorial search to optimize their DNA word codes.

• **Whiplash PCR** (Hagiya, et al [HA+97]) Is a DP-BMC method that uses the end segments of DNA strands to do editing and processing within the interior of the strand. Hagiya, et al [HA+97] showed that Whiplash PCR can be used for SAT problems for a class of Boolean formulas known as μ-formulas, and Winfree [Win98b] extended these techniques to solve general SAT problems. Sakamoto et al [SKK+98] describe how do to finite state transitions using Whiplash PCR, using a graduated scale of melting temperatures to reduce the number of laboratory steps, and also describes implementations of these methods.

• **Decreasing the Volume Used in NP search.** In all these methods, the number of steps grows as a polynomial function of the size of the input, but the volume grows exponentially with the input. For exact solutions of NP complete problems, we may benefit from a more general type of computation than simply brute force search. The molecular computation needs to be general enough to implement sophisticated heuristics, which may result in a smaller search space and volume. For example, Ogihara and Ray [OR97a] proposed a DP-BMC method for decreasing the volume (providing a smaller constant base of the exponential growth rate) required to solve the SAT problem. The difficulty with many of these approaches for NP search is that they initially generate a very large volume containing all possible solutions. An alternative heuristic approach of iteratively refining the solution space. to solve NP search problems has been suggested by Hagiya, et al [HA+97] and Cukras et al [CFL+98], and may in practice give a significant decrease in the volume.

There are a wide variety of further problems that may benefit from the massive parallelism and nano-scale miniaturization available to BMC.

• **Associative Memory using Molecular Parallism** Baum [B95] (also see Lipton [L96]) proposed a parallel memory where DNA strands are used to store memory words, and provided a method for doing associative memory searches using complementary matching. Lipton [Lip98] describes the use of web data bases and associative search within them to do cryptoanalysis.

This idea for associative memory can be extended to allow us to execute operations in parallel, that is to do concurrent word searches. From this follows the concept of a general-purpose molecular computer using DP-BMC. The time and volume efficiency of associative memory searches can be improved by the use of MEMS micro-flow device technology (Gehani and Reif [GR98a]) to segragate pools (micro-Test Tubes) of DNA strands to be searched, and to apply the searches in parallel for each pool.

• **Neural Network Learning and Image Recognition.** Mills, Yurke, and Platzman [MYP98] propose a rather innovative BMC system for error-tolerant learning in a neural network, which is intended to be used for associative matching of images. They use a DP-BMC method for matrix multiplication (Oliver [O96]) to implement the inner products required for neural network training and evaluation, and their proposed BMC system also makes innovative use of DNA chips for I/O.

• **Other Algorithmic Applications of DP-BMC.** DP-BMC may also be used to speed up computations that would require polynomial time on conventional machines: Beigel and Fu [BF97] discuss approximation algorithm for NP search problems, Baum and Boneh discuss DP-BMC methods for executing dynamic programming algorithms, and Oliver [O96] discusses DP-BMC methods for matrix multiplication.

• **Combinatorial Chemistry as NP Searches.** *Combinatorial chemistry* techniques (also known as *diversity* techniques) have been used by biochemists to do combinatorial searches for biological objects with special properties. These techniques were very similar to the use of massive parallelism in BMC to solve NP search problems. Generally, they use recombinant DNA techniques to first construct a large pool of random sequences and then choose elements with specific properties from within the pool. For example, in a widely cited paper, Alper [Al94] discusses the use of diversity techniques for drug discovery. Also, Bartel and Szostak [BS91] constructed a large pool of random sequences and then isolated new ribozymes. Also, Eigen and Rigler [ER94] developed techniques for sorting molecules by closeness metrics. The disciplines of combinatorial chemistry and BMC may benefit by combining some of their techniques. For example, the search space of combinatorial chemistry might be decreased by sophisticated heuristics used in NP search methods.

• **General-purpose Molecular Computers using DP-BMC.** BMC machines using molecular parallelism and providing large memories, are being constructed at Wisconsin [LGCCL+96, CCCFF+97, LTCSC97] and USC [A95, RWBCG+96, ARRW96]. In both projects, a large number of DNA strands are used, where each DNA strand stores multiple memory words. Both these machines will be capable of performing, in parallel, certain classes of simple operations on words within the DNA molecules used as memory. Both projects developed error-resistant word designs. Successful prototyping at moderate scale of either of these machines will be a major experimental milestone in BMC.

The Wisconsin project is employing a surface to immobilize the DNA strands which correspond to the solution space of a NP search problem. Since they are all on the same surface, all DNA strands are operated in a Single Instruction Multiple Data (SIMD) fashion. Their operations on words are restricted to mark, unmark, and destroy operations, which suffice for certain NP search problems. A key challenge in their approach is to provide scaling to a sufficiently large number of DNA strands within the constraints of surface attachment technology.

In contrast, the USC project uses a combination of solution-based and solid support methods, which are used to improve the efficiency of the separation operations. In this method, the computation is done without formation and breaking of covalent bonds. Their operations on words include the Boolean logic operations. All DNA strands within a given test tube are operated on in a SIMD fashion. However, their approach allows splitting of the solution space into separate test tubes, and thus potentially allows for DNA strands to be operated on in a very limited Multiple Instruction Multiple Data (MIMD) fashion, where the number of distinct instructions executed at the same time is limited to the number of test tubes used in parallel. A key challenge in their approach, and the major focus of their effort, is to provide for efficient error-resistant separations.

• **Parallel Arithmetic.** To compete with silicon, it is important to develop the capability of BMC to quickly execute basic operations, such as arithmetic and Boolean operations, that are executed in single steps by conventional machines. Furthermore, these basic operations should be executable in massively parallel fashion (that is executed on multiple inputs in parallel).

Guarnieri and Bancroft [GB96] developed a DNA-based addition algorithm employing successive primer extension reactions to implement the carries and the Boolean logic required in binary addition (similar methods can be used for subtraction). Guarnieri, Fliss, and Bancroft prototyped [GFB96] the first BMC addition operations (on single bits) in recombinant DNA. This experimental work was very significant. However, it suffered from some limitations: (i) only two numbers where added, so it did not take advantage of the massive parallel processing capabilities of BMC and (ii) the outputs were encoded distinctly from the inputs, so it did not allow for repeated operations. Subsequent proposed methods [OGB97, LKSR97, GPZ97] for basic operations such as arithmetic (addition and subtraction) permit chaining of the output of these operations into the inputs to further operations, and to allow operations to be executed in massive parallel fashion. Rubin el al [RKL98] gave an experimental demonstration of a BMC method for chained integer arithmetic. This work also gave one of the first demonstrations in BMC of logically reversible computation. An experimental demonstration of such a method for parallel arithmetic, at large scale, will be a major experimental milestone in BMC. (See also the last subsection of Section 5 for fast local assembly methods for parallel addition and subtraction.)

• **Models for Distributed Molecular Parallelism.**

– **Test Tube and Memory Models.** Lipton [L94] defined the first abstract model of molecular

computation. The elements of his *test tubes* are strings as in the case of DNA. His model allowed a number of operations on test tubes to be executed in one lab step. The subsequent *Memory* model of Adleman [A95] refined the model of Lipton by restricting the set of operations to the following:
Merge: Given tubes T_1, T_2, produce the union $T_1 \cup T_2$.
Copy: Given a tube T_1, produce a tube T_2 with the same contents.
Detect: Given a tube T, say 'yes' if T contains at least one element and say 'no' if it contains none.
Separation: Given a tube T_1 and a word w, produce a tube T_2 with all elements of T_1 that contain w.

Key resource bounds of these abstract models for molecular computations are: *number of steps* required by a molecular algorithm, and *size of the test tube T,* which is the total number of elements of T including replications.

An abstract model of surface-based computation has been developed by [LGCCL+96] (comparable with Lipton and Adelman's models), and it is shown that the surface-based model is also capable of general circuit simulation.

Reif [R95] proposed two further models of molecular computation. The first, the Parallel Associative Memory (PAM) Model, is a very high level model which (i) allows any of the operations for the Memory model of Adleman to be executed in one step, and also (ii) has a Parallel Associative Matching (PA-Match) operation, which provides for the combination of all pairs of DNA strings with subsequences that have a complementary match at a specified location. This PA-Match operation is very similar to the data base join operation.

Reif [R95] also defined a Recombinant DNA (RDNA) Model which is a low level model that allows operations that are abstractions of very well understood recombinant DNA operations and provides a graph representation, known as a *complex*, for the relevant structural properties of DNA. To insure realism, the RDNA model allows complementary pairing of only very short sequences of DNA in constant time. Reif [R95] showed that the PA-Match operation of the PAM model can be simulated in the RDNA model with a slow down which is linear in the pattern match length.

Yokomori and Kobayashi [YK97b] developed a model of BMC based on equality checking, which may be related to the PAM model. Kurtz, Mahaney, Royer, and Simon [KMRS96] formulated a model of BMC which takes into account solution concentrations.

• **Speed-Ups using Molecular Parallelism.** Beaver [BeA95] and Reif [R95] (also Papadimitriou [P95]) independently proved that any linear space bounded sequential computation can be exponentially speeded up by PMC; in particular, they showed that sequential Turing Machine computations with space s and time $2^{O(s)}$ can be simulated by BMC in polynomial time. All their proofs made use of a pointer jumping technique (this pointer jumping technique dates to the 1978 work of Fortune and Wyllie [FW 78], who gave a parallel simulation of a space bounded TM by a PRAM) which required a large volume to implement in BMC. The paper of [R95] proved this speed-up result for the (very high level) PAM model, and then [R95] described in detail its implementation by recombinant DNA operations in the RDNA model. The proof of [B95] used a DNA string-editing operation known as site-directed local mutagenesis (see [WGWZ92], page 192-193, [OP94], page 191-206, and Chapter 5 of [SFM89]) to implement pointer jumping. Khodor and Gifford [KG98] have recently implementated BMC methods using programmed mutagenesis.

• **Molecular PRAMs.** A Parallel Random Access Machine (PRAM) is a parallel machine with a large shared memory. It is CREW if its memory allows concurrent reads and exclusive writes. This same technique of pointer jumping is essential also for Reif's [R95] molecular simulation of a CREW PRAM. Given a CREW PRAM with time bound D, with M memory cells, and processor bound P, [R95] showed that the PRAM can be simulated in the PAM model using t PA-Match operations as well as $O(s \log s)$ PAM operations where $s = O(\log(PM))$ and $t = O(D + s)$. This result immediately implied that in $t = O(D+s)$ PAM steps, one can evaluate and find the satisfying inputs to a Boolean circuit constructable in s space with n inputs, unbounded fan-out, and depth D. Subsequently, Ogihara and Ray [OR97b] obtained a similar result as [R95] for parallel circuit evaluation, implicitly assuming a model similar to the PAM model. (Also see [HA+97] for BMC methods for parallel evaluation of a Boolean μ-formulas.) To allow the PRAM to use shared global memory, we need to do massively parallel message (DNA strand) routing. As a consequence, the volume bounds for this simulation of a PRAM required volume growing at least quadratically with size of the storage of the PRAM. Gehani and Reif [GR98a] propose a MEMS micro-flow device technology that requires a substantial decreased volume to do the massively parallel message routing required for the shared memory operations of the PRAM.

5 The Local Assembly Paradigm

The *local parallelism (LP-BMC)* paradigm for BMC allows operations to be executed in parallel on a given molecule (in contrast to the parallelism where operations are executed in parallel on a large number of distinct molecules but execute sequentially within any given molecule).

Before we describe these local assembly techniques, we first discuss DNA nano-assembly techniques, and some previously known tiling results, which provided the intellectual foundations for local assembly.

• **DNA Nano-Fabrication Techniques.** Feynman [F 61] proposed nano-fabrication of structures of molecular size. Nanotechnology, without use of DNA, is discussed in the texts [CL92, M93].

Nano-fabrication of structures in DNA was pioneered by Seeman (e.g., see [SZC94]) in the 1990s. His work may well be of central importance to the progress of the emerging field of BMC. Seeman and his students such as Chen and Wang nano-fabricated in DNA (see [ZS92, ZS94, SWLQ+96, SQLYL+96, SZDC95] and [SZC94, SC91, SZDWM+94, SQLYL+96]): *2D polygons,* including interlinked squares, and *3D polyhedra,* including a cube and a truncated octahedron. Seeman's ingenious constructions used for basic constructive components:

– *DNA junctions:* i.e., immobile and partially mobile DNA n-armed branched junctions [SCK89],
– *DNA knots:* i.e., ssDNA knots [MDS91, DS92] and Borromean rings[MS97],
– *DNA crossover molecules:* i.e., DX molecules of Fu and Seeman[FS93].

Many of Seeman's constructions used DX molecules for rigidity or dsDNA for partial rigidity. Most of the constructions utilized hybridization in solution, usually followed by ligation. The octahedron used solid-support [S88], to avoid interaction between constructed molecules [ZS92]. See [CRFCC+96, MLMS96] for other work in DNA nano-structures. Recently, Seeman, Liu, et al [SMY+98] constructed from DNA a nanomechanical device capable of controlled movement.

• **Known Tiling Results.** A class of *(domino) tiling problems* were defined by Wang [W61] as follows: we are given a finite set of tiles of unit size square tiles each with top and bottom sides labeled with symbols over a finite alphabet. These labels will be called *pads.* We also specify the initial placement of a specified subset of these tiles, and the borders of the region where tiles must be placed defining the *extent of tiling.* The problem is to place the tiles, chosen with replacement, in all these square regions within the specified borders, so that each pair of vertical abutting tiles have identical symbols on their contacting sides. Let the *size* of the tiling assembly be the number of tiles placed. Berger [B66] (also see Buchi [B62]) proved that given a finite set of tile types, the tiling problem is undecidable if the extent of tiling is infinite. Direct simulations of a single tape deterministic Turing Machines are given in [R71] and [LP81], (pages 296–300). Also, [GJP77] (see [GJ79], page 257) and [LP81](pages 345–348) proved that the domino tiling problem is NP-complete if the extent of tiling is a rectangle of polynomial size. Grunbaum, Branko, and Shepard [GBS87] surveyed these and related results on the complexity of tiling.

• **Computation via Local Assembly.** Winfree [W96] proposed a very intriguing idea: to do these tiling constructions by application of the DNA nano-fabrication techniques of Seeman et al

[SZC94], which may be used for the construction of small DNA molecules that can function as square tiles with pads on the sides. The pads are ssDNA. Recall that if two ssDNA are sticky (i.e., Watson-Crick complementary and $3' - 8'$ and $5' - 3'$ oriented in opposite directions), they may hybridize together at the appropriate conditions into doubly stranded DNA. The assembly of the tiles is due to this hybridization of pairs of matching sticky pads on the sides of the tiles. We will call this innovative paradigm for BMC *unmediated self-assembly* since the computations advance with no intervention by any controllers. The advantages of the unmediated DNA assembly idea of Winfree is potentially very significant for BMC since the computations advance with no intervention by any controllers, and require no thermal cycling. It is a considerable paradigm shift from distributed molecular parallelism, which requires the recombinant DNA steps (which implement the molecular parallelism) to be done in sequence.

To simulate a $1D$ parallel automata or a one tape Turing Machine, Winfree et al [W96, WYS96] proposed self-assembly of $2D$ arrays of DNA molecules, applying the recombinant DNA nanofabrication techniques of Seeman, et al [SZC94], in combination with the tiling techniques of Berger [B66]. Winfree et al [WYS96] then provided further elaboration of this idea to solve a variety of computational problems using unmediated DNA self-assembly. For example, they propose the use of these unmediated DNA assembly techniques to directly solve the NP-complete directed Hamiltonian path problem, using a construction similar to the NP-completeness proof of [GJP77] (see also [GJ79], page 257) for tiling of polynomial size extent. Winfree et al [WYS96] also provided a valuable experimental test validating the preferential pairing of matching DNA tiles over partially non-matching DNA tiles. Winfree [Win98a] made computer simulations of computing by self-assembly of DNA tiles, with a detailed simulation model of the kenetics of annealing during the self assembly of DNA tiles.

Erik Winfree, et al [WLW+98] recently experimentally constructed the first large (involving thousands of individual times) two dimensional arrays of DNA crystals by self-assembly of nearly identical DNA tiles. The tiles consisted of two double-crossovers (DX) which self-assemble into a periodic 2D lattice. They produced spectacular atomic force microscope(AFM) images of these tilings (by insertion of a hairpin sequence into one of the tiles they created 25 nm stripes in the lattice). They also verified the assembly by the use of "reporter" ssDNA sequences. This experiment provided strong evidence of the feasibility of large scaling self-assembly, but it was not in itself computational. LaBean, et al [LWR+99] and [LYK+00] recently designed and experimentally tested in the lab new DNA tiles (TAO adn TAE) which are a rectangular shaped triple crossover molecule with sticky ends on each side that can match with other such tiles and with a "reporter" ssDNA sequence that runs through the tile from lower left to upper right, facilitating output of the tiling computation. Mao, et al [MLR+00] gave the first DNA self-assembly for a (non-trivial) computation. A future major milestones will be to experimentally demonstrate DNA self-assembly of a (possibly non-computational) 3D tiling. A survey of recent progress in DNA tiling and future challenges is given in Reif, et al [RLS00]

• **Assemblies of Small Size and Depth.** To increase the likelihood of success of assembly, Reif [R97] proposed a *step-wise assembly* which provides control of the assembly in distinct steps. The total number of steps is bound by the depth of the assembly. Also, [R97] proposed the use of *frames*, which are rigid DNA nano-structures used to constrain the geometry of the assembly and to allow for the placement of input DNA strands on the boundaries of the tiling assembly. Using these assembly techniques, [R97] proposed LP-BMC methods to solve a number of fundamental problems that form the basis for the design of many parallel algorithms, for these decreased the size of the assembly to linear in the input size and and significantly decreased the number of time steps. For example, the *prefix computation problem* is the problem of applying an associative operation to all prefixes of a sequence of n inputs, and can be used to solve arithmetic problems such as integer addition, subtraction, multiplication by a constant number, finite state automata simulation, and to fingerprint (hash) a string. [R97] gave step-wise assembly algorithms, with linear assembly size and logarithmic time, for the prefix computation problem. As another example, *normal parallel algorithms* [S71, U84, L92] are a large class of parallel algorithms that can be executed in logarithmic time on shuffle-exchange networks (for example DFT, bitonic merge, and an arbitrary fixed permutation of n data elements in logarithmic time). [R97] gave LP-BMC methods for perfect shuffle and pair-wise exchange using a linear size assembly and constant assembly depth, and thus constant time. This allows one to execute normal parallel algorithms using LP-BMC in logarithmic time. Also, this implies a method for parallel evaluation of a bounded degree Boolean circuit in time bounded by the circuit depth times a logarithmic factor. Previously, such operations had been implemented using DP-BMC techniques [R95] in similar time bounds but required a large amount of volume; in contrast the LP-BMC methods of [R97] require very modest volume. All of these LP-BMC algorithms of [R97] can also use DP-BMC to simultaneously solve multiple problems with distinct inputs (e.g. do parallel arithmetic on multiple inputs, or determine satisfying inputs of a circuit), so they are an enhancement of the power of DP-BMC. Jonoska et al [JKS98] describe techniques for solving the Hamiltonian path problem by self assembly of single strand DNA into three dimensional DNA structures representing a Hamiltonian path.

6 The Cellular Processing Paradigm

BMC may make use of microorganisms such as bacteria to do computation. A *cellular processor* is a microorganism such as a bacteria, which does computation via a re-engineered regulatory feedback system for cellular metabolism. The re-engineering involves the insertion of modified regulatory genes. whose DNA has been modified and engineered so that the cell can compute using regulatory feedback systems used in cellular metabolism. This paradigm for BMC was first discussed in a science fiction article of Bear [Bea83]. The recent papers Ji [Ji98] and Kazic [Kaz98] discuss models for doing BMC using cellular processors. Knight and Sussman [KS97] gave a design for logic gates using cellular processing and are planning an experimental demonstration of a cellular processor.

7 The DNA^{2}DNA Paradigm

The field of BMC has restricted its attention mostly to applications which are computational problems, e.g., NP search problems. In this respect, it is still in search of a killer application [R96].

• **Processing Natural DNA.** However, BMC techniques might be ideally suited to solve problems in molecular biology which inherently involve *natural DNA*, that is DNA that is biologically derived (as opposed to artificially synthesized DNA which is coded over a given word alphabet). Lipton, Boneh, and Landweber [LBL 96] considered such a class of problems, including sequencing, fingerprinting and mutation detection. These may well be the killer applications of BMC. An experimental demonstration, at moderate scale, of a BMC method for solving a significant problem in molecular biology with natural DNA inputs, will be a major milestone in BMC.

• **Re-coding DNA.** One interesting approach to use BMC to solve problems concerning natural DNA is to allow natural DNA to be *re-coded*. The natural DNA is re-coded as sequences of encoded n-mers. This re-coding allows the DNA to be then operated in a purely digital manner. The processing of re-coded DNA can then be done by the usual BMC techniques. This is the DNA^{2}DNA paradigm of Landweber and Lipton [LL97].

• **DNA Sequencing.** One possible application considered by [LL97] is DNA sequencing by hybridization [DDSPL+ 93], which is quite different to the enzymatic sequencing techniques commonly used [S88]. Redundant re-coding of n-mers may be used to reduce errors due to incomplete hybridize. These redundant encodings would be constructed and attached to the n-mers using known BMC methods, yielding an encoded array of n-mers providing the DNA sequence information (also see Boneh and Lipton [BL95b] for a quite distinct divide and conquer approach to DNA sequencing).

• **Further Processing Re-coded DNA.** Once natural DNA is re-coded, general BMC methods may be used to speed up many other key applications in biology and medicine [SM97], such as fingerprinting and mutation detection. Re-coded natural DNA derived from many sources can be used to assemble large *wet data bases* containing DNA that encodes data of biological interest, without the problem inherent in I/O to an electronic medium. BMC, with its huge memory capacity, has a considerable advantage over conventional technologies for storing such biological data bases. Once the wet data bases are assembled then we can do further processing using BMC techniques, for example we can do fast associative searches (Baum [B96]) in these wet data bases.

• **Approximate Counting of DNA** Faulhammer, Lipton, and Landweber [FLL98] give a BMC method for estimating the number of DNA strands within a test tube.

8 Conclusion

• **Alternative Paradigms for BMC.** There may well be further alternative paradigms for BMC. For example, Landweber [La96] proposes the use of RNA rather than DNA as the basis of the biotechnology.

• **Comparison of Current BMC with early VLSI.** BMC is a new field, with largely unexplored methodologies. We find it interesting to compare BMC in the later 1990's with the state of VLSI in 1970s, which had (i) multiple enabling technologies which were quickly advancing, (ii) evolving algorithmic paradigms, (iii) lack of simulation models and software for design and simulation of chip designs, and thus (iv) (at the time) high risk. In particular, in the 1970's, the design and fabrication of a VLSI chip was perhaps less an engineering discipline than an art prone to failures, due in part to the lack of developement of (a) exact models for the device physics, (b) software tools for software for design and simulation, and (c) parallel algorithmic design principals.

Through the late 1980's and 1990's, these problems were alleviated for VLSI by the stabilization of the major enabling technology (CMOS), and by major investment by the US government and industry in process modeling and software tools for simulation, allowing for much higher yields in fabrication, and thus considerably lower risk. Also, by this time, there is a mature understanding of parallel algorithmic design principals and high performance architectures (e.g., systolic) for VLSI, thanks to major federal funding programs in these areas. A high pace of improvement in VLSI performance has been sustained for many years, but may slow in the future due to ultimate physical limitations.

BMC now suffers from difficulties similar to those suffered by VLSI in the 1970's. Currently, the design and execution of a BMC experiment in the laboratory is supported with few software tools for design and simulation prior to doing the experiment. (A preliminary version of a Java software tool for simulating BMC has been developed by Gehani, Reif [GR98b].) In spite of an extensive supporting biotechnology, experiments are highly prone to errors. The results described in the the last subsections of Section 2 may alleviate some of these errors, but clearly software tools for design and simulation of BMC experiments need to be developed.

Also, there is at this time no consensus on which methods for doing BMC are the best; as we have seen there are multiple approaches that may have success. While some of the current experiments in BMC are using conventional solution-based recombinant DNA technology, others are employing alternative biotechnology (such as surface attachments). It is also not yet clear which of the paradigms for BMC will be preeminent. To a degree, this diversity of approaches may in itself be an advantage, since it will increase the likelihood of prototyping and establishing successful methodologies.

References

[ALL95] R. A. Adey and A. Lahrmann and C. LeBmollmann, *Simulation and Design of Micro-systems and Micro-structures*, Computational Mechanics Publication, (1995).

[A94] Adleman, L., *Molecular Computation of Solution to Combinatorial Problems*, Science, **266**, 1021–7pt24, (1994).

[A95] Adleman, L., *On Constructing a Molecular Computer*, Dept of CS, U.S.C., Available via anonymous ftp from ftp.usc.edu /pub/csinfo/papers/adleman/molecular_computer·ps, (1995).

[ARRW96] Adleman, L.M., P.W.K. Rothemund, S. Roweis, E. Winfree, *On Applying Molecular Computation To The Data Encryption Standard*, 2nd Annual DIMACS Meeting on DNA Based Computers, Princeton, June, 1996.

[AJPWL+96] Alivisatos, A.P., K.P. Johnsson, X. Peng, T.E. Wilson, C.J. Loweth, M. P. Bruchez Jr., P. G. Schultz, *Organization of 'nanocrystal molecules' using DNA*, Nature, bf 382, 609–611, August 1996.

[Al94] Alper, *Drug discovery on the assembly line*, Science, **264**, 1399–1401, (1994).

[A96] Amenyo, J.-T., *Mesoscopic computer engineering: Automating DNA-based molecular computing via traditional practices of parallel computer architecture design*, Proceedings of the 2nd Annual DIMACS Meeting on DNA Based Computers, June 1996.

[AGH96] Amos, M., A. Gibbons, D. Hodgson, *Error-resistant Implementation of DNA Computations*, 2nd Annual DIMACS Meeting on DNA Based Computers, Princeton University, June 1996.

[AH97] Arita , M. and M. Hagiya, *Joining and Rotating Data with Molecules*, ICEC'97 Special Session on DNA Based Computation, Indiana, April 1997.

[BCGT96] Bach, E., A. Condon, E. Glaser, and C. Tanguay, *Improved Models and Algorithms for DNA Computation*, Proc. 11th Annual IEEE Conference on Computational Complexity, Submitted (by invitation) to: J. Computer and System Sciences, May 1996.

[BS91] Bartel, D. and J. Szostak, *Isolation of new ribozymes from a large pool of random sequences*, Science, **261**, 1411–1418, (1991).

[B68] Batcher, K. *Sorting Networks and their applications*, Spring Joint Computer Conference, **32**, 307–314, AFIPS Press, Montvale, N. J., (1968).

[B95] Baum, E. B. How to build an associative memory vastly larger than the brain, Science pp 583-585, Vol. 268, (April 28,1995).

[B96] Baum, E. B. *DNA Sequences Useful for Computation*, 2nd Annual DIMACS Meeting on DNA Based Computers, Princeton University, June 1996.

[BB96] Baum, E. B. and D. Boneh, *Running dynamic programming algorithms on a DNA computer*, 2nd Annual DIMACS Meeting on DNA Based Computers, Princeton, June, 1996.

[Bea83] G. Bear, *Blood Music*, Analog, (June, 1983). Also, appearing in Nanodreams, (Ed. by E Elliott), Baen Pub., (1995). Also, appearing as a full novel as *Blood Music*, Ace Pub., (1985). *Blood Music*, Analog, (june, 1983).

[Be94] Beaver, D. *Factoring: The DNA Solution*, Advances in Cryptology, Asia Crypt94 Proceedings, Lecure Notes in Computer Science, (1994), Springer Verlag, (http:// www.cse.psu.edu/ beaver/ publications/ pubindex.html),

[BeB95] Beaver, D. *Computing with DNA*, J. Comp. Biol., **2**, 1–7, (1995).

[BeA95] Beaver, D. *A Universal Molecular Computer*, revised as *Molecular Computing*, Penn State University Technical Memo CSE-95-001, Pond Lab, http:// www.cse.psu.edu/ beaver/ publications/ pubindex.html, (1995).

[BF97] Beigel, R. and Bin Fu, *Molecular Approximation Algorithm for NP Optimization Problems*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[B65] Beneš, V. *Mathematical Theory of Connecting Networks and Telephone Traffic*, Academic Press, New York, NY (1965).

[B66] Berger, R. *The Undecidability of the Domino Problem*, Memoirs of the American Mathematical Society, **66**, (1966).

[B97] Blumberg, A.J. *Parallel Computation on a DNA Substrate*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[BDL95] Boneh, D., C. Dunworth, R. Lipton, *Breaking DES Using a Molecular Computer*, Princeton CS Tech-Report number CS-TR-489-95 (1995).

[BDLS95] Boneh, D., C. Dunworth, R. Lipton, J. Sgall, *On the Computational Power of DNA*, Princeton CS Tech-Report number CS-TR-499-95 (1995). Also published in Discrete Applied Math, (Dec 96)

[BL95a] Boneh, D., and R. Lipton, *Making DNA Computers Error Resistant*, Princeton CS Tech-Report CS-TR-491-95 Also in 2nd Annual DIMACS Meeting on DNA Based Computers,, Princeton University, June 1996.

[BL95b] Boneh, D., and R. Lipton, *A Divide and conquer approach to DNA sequencing*, Princeton University, 1996.

[BKP90] Brooks, C., M. Karplus, M. Pettitt, *Proteins, A Theoretical Perspective of Dynamics, Structure & Thermodynamics*, Adv. Chem. Phys., 71 (1988).

[B62] Buchi, J.R *Turing Machines and the Entscheidungsproblem*, Mathematiche Annalen, **148**, 201–213, (1962).

[CCCFF+97] Cai, W., A. Condon, R.M. Corn, Z. Fei, T. Frutos, E. Glaser, Z. Guo, M.G. Lagally, Q. Liu, L.M. Smith, and A. Thiel, *The Power of Surface-Based Computation*, Proc. First International Conference on Computational Molecular Biology (RECOMB97), January, 1997.

[CRFCC+96] Cai, W., E. Rudkevich, Z. Fei, A. Condon, R. Corn, L.M. Smith, M.G. Lagally, *Influence of Surface Morphology in Surface-Based DNA Computing*, Submitted to the 43rd AVS National Symposium, Abstract No. BI+MM-MoM10, (1996).

[CCTKS88] J.-H. Chen, M.E.A. Churchill, T.D. Tullius, N.R. Kallenbach, N.C. Seeman, *Construction and Analysis of Monomobile DNA Junctions*, Biochemistry, **27**, (1988).

[CRWEC95] Chen, J. , C.A. Rauch, J.H. White, P.T. Englund, N.R. Cozzarelli, em The Topology of the Kinetoplast DNA Network, rm Cell, **80**, 61–69, January 1995.

[CW97] Chen, J., and D. Wood, *A New DNA Separation Technique with Low Error Rate*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[Con98] Conrad, M., *Molecular and evolutionary computation: The tug of war between context freedom and context sensitivity*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[CZ97] Conrad, M. and K.-P. Zauner, *Design for a DNA Conformational Processor*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[CL92] Crandall, B. and J. Lewis (eds.), *Nanotechnology*, MIT Press, (1992).

[CFKP96] E. Csuhaj-Varju, R. Freund, L. Kari, and G. Paun, *DNA Computing Based on Splicing: Universality Results*, Proceedings of 1st Annual Pacific Symposium on Biocomputing, Hawaii, (L.Hunter, T.Klein, eds.), World Scientific Publ., Singapore, 179–190. (1996).

[CFL+98] Cukras, A., D. Faulhammer, R. Lipton, L. Landweber, *Chess games: A model for RNA-based computation*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[CH89] K. Culik II and T. Harju, *The regularity of splicing systems and DNA*, Proc. ICALP'89, Lec. Notes. in C.S., **372**, 222–233, (1989).

[DMGFS96] Deaton, R., R.C. Murphy, M. Garzon, D.R. Franceschetti, and S.E. Stevens, Jr., *Good encodings for DNA-based solutions to combinatorial problems*, Proceedings of the 2nd Annual DIMACS Meeting on DNA Based Computers, June 1996.

[DMRGF+97] Deaton, R., R.C. Murphy, J.A. Rose, M. Garzon, D.R. Franceschetti, and S.E. Stevens, Jr., *A DNA Based Implementation of an Evolutionary Search for Good Encodings for DNA Computation*, ICEC'97 Special Session on DNA Based Computation, Indiana, April 1997.

[DHK96] Delcher, A. L., L. Hood, R.M. Karp, *Report on the DNA/Biomolecular Computing Workshop*, June 1996.

[DHS97] Deputat, M., G. Hajduczok, E. Schmitt, *On Error-Correcting Structures Derived from DNA*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[DDSPL+ 93] Drmanac, R, S. Drmanac, Z. Strezoska, T. Paunesku, I. Labat, M. Zeremski, J. Snoddy, W. K. Funkhouser, B. Koop, L. Hood, and R. Crkenjakov *DNA Sequence Determination by Hybridize: A Strategy for Efficient Large-Scale Sequencing*, Science, **260**, 1649–1652, (1993).

[DS92] Du, S.M., and N.C. Seeman, *The Synthesis of a DNA Knot Containing both Positive and Negative Nodes*, J. Am. Chem. Soc., **114**, 9652–9655, (1992).

[DZS92] Du, S.M., S. Zhang and N.C. Seeman, *DNA Junctions, Antijunctions and Mesojunctions*, Biochem., **31**, 10955–7pt963, (1992).

[ER94] Eigen, M., and R. Rigler, *Sorting Single Molecules - applications to diagnostic and evolutionary biotechnology*, Proc. of the National Academy of Science, **91**, 5740–8747, (1994).

[EC] Eisenberg and Crothers, *Physical Chemistry with applications to the life sciences.*

[E97] Eng, T. *Linear DNA Self-Assembly with Hairpins Generates the Equivalent of Linear Context-Free Grammars*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[Eng98] Eng, T., *On solving a 3CNF-Satisfiability with an in vivo algorithm*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[ES97] Eng, T., and B.M. Serridge, *A Surface-Based DNA Algorithm for Minimal Set Cover*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[ER82] Engler, M., and C. Richardson, *The Enzyme*, (P. Boyer, ed.) Academic Press, 3–29, (1982).

[FF97] Faulhammer, D. and Michael Famulok, *In Vitro Selection and Characterization of DNA Enzymes*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[FLL98] Faulhammer, D., R. Lipton, L. Landweber, *Counting DNA: Estimating the complexity of a test tube of DNA*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[F 61] Feynman, R. There's Plenty of Room at the Bottom, Miniaturization (D. Gilbert, ed.), Reinhold, 282–296, (1961). Reprinted in B.C. Crandall and J. LKewis, eds., Nanotechnology: Research and Perspectives, Cambridge: The MIT Press, p. 360, (1992).

[FW 78] Fortune, S. and J. Wyllie, *Parallelism in random access machines*, Proc. 10th Annual ACM S.T.O.C., San Diego, CA, 114–118, (1978).

[F97] Freund, R. *Test Tube Systems with Controlled Applications of Rules*, ICEC'97 Special Session on DNA Based Computation, Indiana, April, 1997.

[FKP98] R.Freund, L.Kari, G.Paun, *DNA computing: the existence of universal computers*, to appear in Theory of Computer Science, (1998).

[FPRS97] Freund, R., G. Paun, G. Rozenberg, A. Salomaa *Watson-Crick Finite Automata*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[FTCSC97] Frutos, A.G., A.J. Thiel, A.E. Condon, L.M. Smith, R.M. Corn, *DNA Computing at Surfaces: 4 Base Mismatch Word Design*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[FB98] Fu, B., R. Beigel, *Length bounded molecular computing*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[FBZ98] Fu, B., R. Beigel, F. Zhou, *An $O(2^n)$ volume molecular algorithm for Hamiltonian Path*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[FS93] Fu, T.-J., and N.C. Seeman, *DNA Double Crossover Structures*, Biochemistry, **32**, 3211–3220, (1993).

[GGM97] Gao, Y. M. Garzon, R.C. Murphy, J.A. Rose, R. Deaton, D.R. Franceschetti, S.E. Stevens Jr., *DNA Implementation of Nondeterminism*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[GJ79] Garey, M. R., and D. S. Johnson *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H Freeman and Company, page 257, (1979).

[GJP77] Garey, M.R., D. S. Johnson, and C. H. Papadimitriou, unpublished manuscript, (1977).

[GDNMF97] Garzon, M., R. Deaton, P. Neathery, R.C. Murphy, D.R. Franceschetti, S.E. Stevens Jr., *On the Encoding Problem for DNA Computing*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[GJ98] Garzon, M., N. Jonoska, *The bounded complexity of DNA computing*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998). To appear in DNA Based Computers, IV, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, (ed. H. Rubin), American Mathematical Society, 1999. Also to appear in a special issue of Biosystems, 1999.

[GLR99] A. Gehani, T. H. LaBean, and J.H. Reif, DNA-based Cryptography, 5th DIMACS Workshop on DNA Based Computers, MIT, June, 1999. To appear in DNA Based Computers, V, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, (ed. E. Winfree), American Mathematical Society, 2000. http://www.cs.duke.edu/~reif/paper/DNAcypt/crypt.ps

[GR98a] Gehani, A., J. Reif, *Micro flow bio-molecular computation*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998). DNA Based Computers, IV, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, (ed. H. Rubin), American Mathematical Society, 1999. Appeared in a special issue of Biosystems, Journal of Biological and Informational Processing Sciences, Vol. 52, Nos. 1-3, (ed. By L. Kari, H. Rubin, and D. H. Wood), pp 197-216, (1999). www.cs.duke.edu/~reif/paper/geha/microflow.ps

[GR98b] Gehani, A., J. Reif, *A Simulation System for Bio-molecular Computation*, to appear.

[G94] Gifford, D. *On the Path to Computing with DNA*, Science, **266**, 993–994, November, 1994.

[GKG+98] Gloor, G., L. Kari, M. Gaasenbeek, S. Yu, *Towards a DNA solution to the shortest common superstring problem*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998). Also, Proceedings of IEEE'98 International Joint Symposia on Intelligence and Systems, Rockville, MD, 140–145, (May 1998).

[GFBCL+96] Gray, J. M. , T. G. Frutos, A.M. Berman, A.E. Condon, M.G. Lagally, L.M. Smith, R.M. Corn, *Reducing Errors in DNA Computing by Appropriate Word Design*, University of Wisconsin, Department of Chemistry, October 9, 1996.

[GBS87] Grunbaum, S., Branko, and G.C. Shepard *Tilings and Patterns*, H Freeman and Company, **Chapter 11**, (1987).

[GB96] Guarnieri, F., and C. Bancroft, *Use of a Horizontal Chain Reaction for DNA-Based Addition*, Proceedings of the 2nd Annual DIMACS Meeting on DNA Based Computers., June 10-12, 1996, American Mathematical Society, Providence, RI (in press), (1996).

[GFB96] Guarnieri, F., Fliss, M., and C. Bancroft, *Making DNA add*, Add. Science, **273**, 220–223, (1996).

[GPZ97] Gupta, V., S. Parthasarathy, M.J. Zaki, *Arithmetic and Logic Operations with DNA*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[HA+97] Hagiya, M., M. Arita, D. Kiga, K. Sakamoto, and S. Yokoyama, *Towards Parallel Evaluation and Learning of Boolean μ-Formulas with Molecules*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[HG97] Hartemink, A.J., D.K. Gifford *Thermodynamic Simulation of Deoxyoligonucleotide Hybridize for DNA Computation*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[HGL98] Hartemink, A., David Gifford, J. Khodor, *Automated constraint-based nucleotide sequence selection for DNA computation*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[H87] Head, T., *Formal language theory and DNA: an analysis of the generative capacity of specific recombinant behaviors*, Bull. Math. Biology, **49**, 737–759, (1987).

[H92] Head, T., *Splicing schemes and DNA*, In: *Lindenmayer Systems: Impacts on Theoretical Computer Science, Computer Graphics, and Developmental Biology*, Ed. by G.Rozenberg and A.Salomaa, Springer-Verlag, 371-383, (1992). Also appears in: *Nanobiology*, **1**, 335–342, (1992).

[H97] Head, T., *Splicing System and Molecular Processes*, ICEC'97 Special Session on DNA Based Computation, Indiana, April, 1997.

[HPP96] Head, T., G. Paun, and D. Pixton, *Language Theory and Molecular Genetics-generative mechanisms suggested by DNA recombination*, A chapter in: *Handbook of Formal Language Theory*, Springer-Verlag, (1996 or to appear).

[Iku96] K. Ikuta *3D Micro Integrated Fluid System Toward Living LSI*, International Workshop on Artificial Life (1996).

[J92] JáJá, J. *An Introduction to Parallel Algorithms*, Addison Wesley, (1992).

[JKS98] Jonoska, N, S. Karl, M. Saito, *Three dimensional DNA structures in computing*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[Ji98] Ji, S., *The cell as a DNA-based molecular computer*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[JK96] Jonoska, N., and S.A. Karl, *A Molecular Computation of the Road Coloring Problem*, 2nd Annual DIMACS Meeting on DNA Based Computers, Princeton University, June 1996.

[JK97a] Jonoska, N., and S.A. Karl, *Ligation Experiments in Computing with DNA*, ICEC'97 Special Session on DNA Based Computation, Indiana, April 1997.

[JK97b] Jonoska, N., and S.A. Karl, *Creating 3-Dimensional Graph Structures with DNA*, 3nd Annual DIMACS Meeting on DNA Based Computers, University of Pens., (June 1997).

[KCL96] Kaplan, P., G. Cecchi, and A. Libchaber, *DNA based molecular computation: Template-template interactions in PCR*, The 2nd Annual Workshop on DNA Based Computers, American Mathematical Society, To appear, (1996).

[KTL97] Kaplan, P., D. Thaler, A. Libchaber, *Parallel Overlap Assembly of Paths Through a Directed Graph*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[Kar96] L.Kari, *DNA computers: tomorrow's reality* , Tutorial in the Bulletin of EATCS, no.59, pp.256-266, (1996).

[Kar97A] L.Kari, *DNA computing based on insertions and deletions*, Proceedings of the conference Conceptual tools for understanding dynamics in biological systems, London, 1996. In COENOSES, C.E.T.A. Gorizia, Italy, N. Kenkel, ed., Vol. 12, pp. 2–3, 89-95, (1997).

[Kar97B] L.Kari, *DNA computing - the arrival of biological mathematics*, The Mathematical Intelligencer, vol.19, **2:** 9–22, (1997).

[Kar98] L.Kari, *Computing with DNA*, In Computer Methods in Molecular Biology, (S.Misener, S.Krawetz, Eds.), in Methods in Molecular Biology series, Humana Press. To appear, (1998).

[KPRS98] L.Kari, G.Paun, G.Rozenberg, A.Salomaa, S.Yu, *DNA computing, sticker systems, and universality*, Acta Informatica, **35:** 401420, (1998).

[KPTY97] Kari, L., G. Paun, G. Thierrin, Sheng Yu, *At the Crossroads of DNA Computing and Formal Languages: Characterizing Recursively Enumerable Languages Using Insertion-Deletion Systems*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[KarS97] L.Kari, Y.Sakakibara., *DNA Computers* Journal of the Institute of Electronics, Information and Communication Engineers, vol.80, no.9, 1997, pp. 935-939 (in Japanese), (1997) [Kaz98] Kazic, T., *After the Turing machine: A metamodel for molecular computing*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[KG97] Khodor, J., and David K. Gifford, *The Efficiency of Sequence-Specific Separation of DNA Mixtures for Biological Computing*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[KG98] Khodor, J., D. Gifford, *Design and implementation of computational systems based on programmed mutagenesis*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[KK97] Kim, S.M., and Kyungpook, *Identifying Genetically Spliced Languages*, ICEC'97 Special Session on DNA Based Computation, Indiana, April, 1997.

[KS97] Knight,T. F., G.J. Sussman, *Cellular Gate Technology*, MIT Artificial Intelligence Labratory, July 1997.

[KMRS96] Kurtz, S.A., S.R. Mahaney, J.S. Royer, J. Simon, *Active Transport in Biological Computing*, 2nd Annual DIMACS Meeting on DNA Based Computers, Princeton University, June 1996.

[LYK+00] LaBean, T. H., Yan, H., Kopatsch, J., Liu, F., Winfree, E., Reif, J.H. and Seeman, N.C., The construction, analysis, ligation and self-assembly of DNA triple crossover complexes, J. Am. Chem. Soc. 122, 1848-1860 (2000). www.cs.duke.edu/∼reif/paper/DNAtiling/tilings/JACS.pdf

[LWR+99] LaBean, T. H., E. Winfree, J. H. Reif, Experimental Progress in Computation by Self-Assembly of DNA Tilings, 5th International Meeting on DNA Based Computers(DNA5), MIT, Cambridge, MA, (June, 1999). To appear in DIMACS Series in Discrete Mathematics and Theoretical Computer Science, ed. E. Winfree, to appear American Mathematical Society, 2000. http://www.cs.duke.edu/∼thl/tilings/labean.ps

34.

[LF80] Ladner, R.E., and M.J. Fischer, *Parallel Prefix Computation*, JACM, **27(4)**:831–838, (1980).

[La96] Landweber, L.F., *RNA Based Computing*, American Mathematical Society, R. J. Lipton and E. B. Baum, eds., 2nd Annual DIMACS Meeting on DNA Based Computers, DIMACS Workshop, Princeton, June, 1996

[L97] Landweber, L., *In vitro evolution of a novel RNA ligase ribozyme from a large pool of random sequences*, to appear, (1997).

[LG 93] Landweber, L.F. and W. Gilbert, *RNA editing as a source of genetic variation*, Nature, **363**, 179–182, (1993).

[LK98] Landweber, L., L. Kari, *The evolution of cellular computing: Nature's solution to a computational Problem*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998). Also, Proceedings of the 3rd Annual Genetic Programming Conference, Morgan Kaufmann Publishers, San Francisco, pp.700-708, (July 22-25, 1998).

[LG 94] Landweber, L.F. and W. Gilbert, *Phylogenetic analysis of RNA editing: A primitive genetic phenomenon*, Proc. Natl. Acad. Sci., **91**, 918–921, (1994).

[LL97] Landweber, L.F. and R. Lipton, *DNA 2 DNA Computations: A Potential 'Killer App'?*, 3nd Annual DIMACS Meeting on DNA Based Computers, University of Pens., (June 1997).

[LR97] Laun, E., and K.J. Reddy, *Wet Splicing Systems*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[LKSR97] Leete, T.H., J. Klein, J.S. Salem, and H. Rubin, *Bit Operations Using a DNA Template*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[LS+96] Leete, T.H., M.D. Schwartz, R.M. Williams, D.H. Wood, J.S. Salem, and H. Rubin, *Massively Parallel DNA Computation: Expansion of Symbolic Determinants*, 2nd Annual DIMACS Meeting on DNA Based Computers, Princeton, June, 1996, Also, **American Mathematical Society**, To appear, (1997). (Also U.S. Patent Application, 1996).

[L92] Leighton, F.T. *Introduction to Parallel Algorithms and Architectures*, Morgan Kaufmann Press, San Mateo, CA, Chapter 3, (1992).

[LP81] Lewis, H.R., and C.H. Papadimitriou *Elements of the Theory of Computation*, Prentice-Hall, pages 296–300 and 345–348 (1981).

[Li98] Z. Li, Z., *Algebraic properties of DNA operations*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[LYQS96] Li, X.J., X.P. Yang, J. Qi, and N.C. Seeman, *Antiparallel DNA Double Crossover Molecules as Components for Nanoconstruction*, J. Am. Chem. Soc., **118**, 6131–6140, (1996).

[L94] Lipton, R. *Speeding Up Computations via Molecular Biology*, Princeton University Draft, (1994).

[L95] Lipton, R.J. *DNA Solution of Hard Computational Problems*, Science, **268**, 542–845, (1995).

[L96] Lipton, R.J. *DNA Computations Can Have Global Memory*, unpublished manuscript, April 1996.

[Lip98] Lipton, R., *A memory based attack on cryptosystems with application to DNA computing*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[LBL 96] Lipton, R.J., D. Boneh, and L. Landweber, *Analog DNA Based Computation*, in preparation. (1996).

[LFW+98] Liu, Q., A. Frutos, L. Wang, A. Thiel, S. Gillmor, T. Strother, A. Condon, R. Corn, M. Lagally, L. Smith, *Progress towards demonstration of a surface based DNA computation: A one word approach to solve a model satisfiability problem*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[LGCCL+96] Liu, Q., Z. Guo, A.E. Condon, R.M. Corn, M.G. Lagally, and L.M. Smith, *A Surface-Based Approach to DNA Computation,*, Proc. 2nd Annual Princeton Meeting on DNA-Based Computing, June 1996.

[LTCSC97] Liu, Q., A.J. Thiel, A.G. Frutos, R.M. Corn, L.M. Smith, *Surface-Based DNA Computation: Hybridize and Destruction*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[ADL+98] Manca, V., C. Martin-Vide, G. Paun, *New computing paradigms suggested by DNA computing by carving*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[MLR+00] Mao, C., T.H. LaBean, J. H. Reif, and N.C. Seeman, An Algorithmic Self-Assembly, Nature, Sept 28, (2000). www.cs.duke.edu/∼reif/paper/SELFASSEMBLE/AlgorithmicAssembly.pdf

[MR98] Margenstern, M., Y. Rogozhin, *A universal time-varying distributed H system of degree 2*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[MEBH92] A. Manz, C. S. Effenhauser, N. Burggraf, D. J. Harrison, K. Seiler, and K. Fluri, *Electro-osmotic Pumping and Electro-osmotic Pumping and Electro-phoretic Separations for Miniaturized Chemical Analysis Systems*, Journal of Micro-mechanics and Micro-engineering, (1992).

[MH87] McCammon, J., and S. Harvey, *Dynamics of Proteins and Nucleic Acids*, Cambridge University Press, (1987).

[M93] Merkle, R. Nanotechnology, **4** 21, (1993).

[M97] Mihalache, V. *Prolog Approach to DNA Computing*, ICEC'97 Special Session on DNA Based Computation, Indiana, April, 1997.

[MYP98] Mills, A., B. Yurke, P. Platzman, *Error-tolerant massive DNA neural-network computation*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[M96] Mir, K.U. *A Restricted Genetic Alphabet for DNA Computing*, 2nd Annual DIMACS Meeting on DNA Based Computers,, Princeton University, June 1996.

[MLMS96] Mirkin, C.A., R.L. Letsinger, R.C. Mucic, J.J. Storhoff, *A DNA-based Method for Rationally Assembling Nanoparticles Into Macroscopic Materials*, Nature, **382**, 607–611, August 1996.

[MS97] Mao, C., and N.C. Seeman, *Construction of Borromean Rings from DNA*, Nature, **386**(6621), 137–138, (March,1997).

[M et al, 2000] Mao, C., T.H. LaBean, J. H. Reif, and N.C. Seeman, An Algorithmic Self-Assembly, Nature, Sept 28, (2000). www.cs.duke.edu/∼reif/paper/SELFASSEMBLE/AlgorithmicAssembly.pdf

[MoS97] Morimoto, N., M.A.A. Suyama, *Solid Phase DNA Solution to the Hamiltonian Path Problem*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[MDS91] Mueller, J.E., S.M. Du, and N.C. Seeman, *The Design and Synthesis of a Knot from Single-Stranded DNA*, J. Am. Chem. Soc., **113**, 6306–6308, (1991).

[MDFS97] Murphy, R.C., R. Deaton, D.R. Franceschetti, S.E. Stevens, *A New Algorithm for DNA Based Computation*, ICEC'97 Special Session on DNA Based Computation, Indiana, April 1997.

[OR97a] Ogihara, M., and A. Ray, *Breadth first search 3SAT algorithms for DNA computer*, Technical Report TR-629, Department of Computer Science, University of Rochester, (July 1996).

[OR97b] Ogihara, M., and A. Ray, *Simulating Boolean circuits on a DNA computer*, 1st Annual International Conference On Computational Molecular Biology (RECOMB97), Santa Fe, New Mexico, January 1997.

[OR97c] Ogihara, M., and A. Ray, *DNA-Based Parallel Computation by 'Counting'*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[OP94] Old, R., and S. Primrose, *Principles of Gene Manipulation, An Introduction to Genetic Engineering*, Blackwell Scientific Publications, Fifth Edition, (1994).

[O96] Oliver, J.S. *Computation With DNA-Matrix Multiplication*, 2nd Annual DIMACS Meeting on DNA Based Computers, Princeton, June, 1996.

[OGB97] Orlian, M., F. Guarnieri, C. Bancroft, *Parallel Primer Extension Horizontal Chain Reactions as a Paradigm of Parallel DNA-Based Computation*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[P95] Papadimitriou, C. personal communication, (1995).

[P96a] Paun, G., *On the splicing operation*, Discrete Applied Math., **70**, 57–79, (1996).

[P96b] Paun, G., *Five Universal DNA Computing Models Based on the Splicing Operation*, to appear, (1996).

[P97] Paun, G., *Computing by Splicing: Programmed and Evolving Splicing Systems*, ICEC'97 Special Session on DNA Based Computation, Indiana, April, 1997.

[PRS96] Paun, G., Rozenberg, and A. Salomaa, *Computing by splicing,*, Theor. Computer Sci., **168**, 321–336, (1996).

[Pi95] Pixton, D., *Linear and circular splicing systems*, Proc. 1st Intn. Symp. on Intelligence in Neural and Biological Systems, IEEE Press, 181-188, (1995).

[Pi96] Pixton, D., *Regularity of splicing languages*, Discrete Applied Math., **69**, 101–124, (1996).

[Pi97] Pixton, D., *Splicing systems and AFL theory*, (to be submitted shortly).

[PP97] Plum, G.E., and D. S. Pilch, *Nucleic Acid Hybridize: Triplex Stability and Energetics.*, Annu. Rev. Biophys. Biomol. Struct. , **24:**, 319–350, (February 1997).

[Po97] Pool, R. *Dr. Tinkertoy*, Discover, **18:2**, 50–87, (February 1997).

[QYS96] Qi, J., X.J. Li, X.P. Yang, and N.C. Seeman, *Ligation of triangles built from bulged 3-arm DNA branched junctions*, J. Am. Chem. Soc., **v118:26**, 6121–6130, (July, 1996).

[R93] Reif, J. (ed.), *Synthesis of Parallel Algorithms*, Morgan Kaufmann, (1993).

[R95] Reif, J., *Parallel Molecular Computation: Models and Simulations*, Seventh Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA95), ACM, Santa Barbara, 213–223, June 1995. Published in **Algorithmica**, special issue on Computational Biology, 25:142-175, 1998. This paper in postscript, can be found at $http : //www.cs.duke.edu/\tilde{r}eif/paper/Molecular.ps$ and figures can be found at $http : //www.cs.duke.edu/\tilde{r}eif/paper/mole.fig.ps$.

[R97] Reif, J.H., *Local Parallel Biomolecular Computation*, Postscript versions of this paper and its figures are at http://www.cs.duke.edu/~reif/paper/Assembly.ps and http://www.cs.duke.edu/~reif/paper/Assembly.fig.ps, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[R98] Reif, J.H., *Alternative Computational Models: A Comparison of Biomolecular and Quantum Computation*, (A postscript preprint of this paper can be found at http://www.cs.duke.edu/~ reif/paper/altcomp.ps.), invited paper,18th International Conference on Foundations of Software Technology and Theoretical Computer Sceince (FST&TCS98), (December, 1998).

[RLS00] J.H. Reif, T. H. LaBean, and Seeman, N.C.,Challenges and Applications for Self-Assembled DNA Nanostructures, Invited paper, Sixth International Meeting on DNA Based Computers (DNA6), DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Leiden, The Netherlands, (June, 2000) ed. A. Condon. To be published by Springer-Verlag as a volume in Lecture Notes in Computer Science, (2000). http://www.cs.duke.edu/~reif/paper/SELFASSEMBLE/selfassemble.ps

[RL00] J.H. Reif and T. H. LaBean, Computationally Inspired Biotechnologies: Improved DNA Synthesis and Associative Search Using Error-Correcting Codes and Vector-Quantization, Sixth International Meeting on DNA Based Computers (DNA6), DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Leiden, The Netherlands, (June, 2000) ed. A. Condon. To be published by Springer-Verlag as a volume in Lecture Notes in Computer Science, (2000). www.cs.duke.edu/~reif/paper/Error-Restore/Error-Restore.ps

[RE97] Robertson, M.P., and Andrew D. Ellington, *New Directions in Nucleic Acid Computing: Selected Ribozymes that Can Implement Re-Write Rules*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[R71] Robinson, R.M. *Undecidablility and Nonperiodicity for Tilings of the Plane*, Inventiones Mathematicae, **12**, 177–209, (1971).

[RDGS97] Rose, J.A., R. Deaton, M. Garzon, and S.E. Stevens Jr., *The Effect of Uniform Melting Temperatures on the Efficiency of DNA Computing*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[Ro95] Rothemund, P.W.K. *A DNA and restriction enzyme implementation of Turing Machines*, manuscript available at http://www.ugcs.caltech.edu/ pwkr/oett.html, (1995).

[RW95] Rooβ, D., and K.W. Wagner, *On the power of Bio-Computers*, unpublished manuscript.

[RWBCG+96] Roweis, S., E. Winfree, R. Burgoyne, N.V. Chelyapov, M.F. Goodman, P.W.K. Rothemund, L. M. Adleman, *A Sticker Based Architecture for DNA Computation*, 2nd Annual DIMACS Meeting on DNA Based Computers, Princeton University, June1996, Also as Laboratory for Molecular Science, USC technical report *A Sticker Based Model for DNA Computation*, May 1996.

[RS97] Rozenberg, G., and A. Salomaa, ICEC'97 Special Session on DNA Based Computation, Indiana, April 1997.

[R96] Rubin, H. *Looking for the DNA killer app.*, Nature, **3**, 656–658, (1996).

[RKL98] Rubin, H., J. Klein, T. Leete, *A biomolecular implementation of logically reversible computation with minimal energy dissipation*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[SF97] Sakakibara, Y., and C. Ferretti, *Splicing on Tree-like Structures*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[SKK+98] Sakamoto, K., D. Kiga, K. Komiya, H. Gouzu, S. Yokoyama, S. Ikeda, H. Sugiyama, M. Hagiya, *State transitions by molecules*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[SFM89] Sambrook, J., E. Fritsch, and T. Maniatis, Molecular Cloning: A Laboratory Manual, Cold Spring Harbor Laboratory Press, NY, Vol. 1,2,3 (1989).

[S82] Seeman, N.C. *Nucleic Acid Junctions and Lattices*, J. Theor. Biol., **99**, 237-247, (1982).

[S85] Seeman, N.C. *Macromolecular Design, Nucleic Acid Junctions and Crystal Formation*, Journal of Biomolecular Structure and Dynamics, **3**, 1-34, (1985).

[SC91] Seeman, N. C., and J. Chen, *Synthesis from DNA of a molecule with the connectivity of a cube*, Nature, **350**, 631–633, (1991).

[SCDMZ+93] Seeman, N. C., J. Chen, S.M. Du, John E. Mueller, Yuwen Zhang, Tsu-Ju Fu, Yinli Wang, Hui Wang, Siwei Zhang, *Synthetic DNA knots and catenanes*, New Jour. of Chemistry, **17**, 739–755, (1993).

[SCK89] Seeman, N. C., J.-H. Chen, N.R. Kallenbach, *Gel electrophoretic analysis of DNA branched junctions*, Electrophoresis, **10**, 345–354, (1989).

[SMY+98] Seeman, N., F. Liu, C. Mao, X. Yang, L. Wenzler, E. Winfree, *DNA nanotechnology: Control of 1-D and 2-D arrays and the construction of a nanomechanical device*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[SQLYL+96] Seeman, N. C., J. Qi, X. Li, X. Yang, N.B. Leontis, B. Liu, Y. Zhang, S.M. Du, and J. Chen, *The control of DNA structure: From topological modules to geometrical modules*, Modular Chemistry, J. Michl, ed., Kluwer, To appear, (1996).

[SWLQ+96] Seeman, N. C., H. Wang, B. Liu, J. Qi, X. Li, X. Yang, F. Liu, W. Sun, Z. Shen, R. Sha, C. Mao, Y. Wang, S. Zhang, T.-J. Fu, S. Du, J. E. Mueller, Y. Zhang, and J. Chen, *The Perils of Polynucleotides: The Experimental Gap Between the Design and Assembly of Unusual DNA Structures*, The 2nd Annual Workshop on DNA Based Computers, American Mathematical Society, June 1996.

[SZC94] Seeman, N. C., Y. Zhang, and J. Chen, *DNA nanoconstructions*, J. Vac. Sci. Technol., **12:4**, 1895–1905, (1994).

[SZDC95] Seeman, N. C., Y. Zhang, S.M. Du, and J. Chen, *Construction of DNA polyhedra and knots through symmetry minimization*, Supramolecular Stereochemistry, J. S. Siegel, ed., 27–32, (1995).

[SZDWM+94] Seeman, N. C., Y. Zhang, S. Du, H. Wang, J.E. Mueller, and J. Chen, *The control of DNA structure and topology: An overview*, Mat. Res. Soc. Symp. Proc., **356**, 57–66, (1994). DNA, Molecular Biology

[SM97] Setubal, J., and J. Meidanis *Introduction to Computational Molecular Biology*, PWS Pub. Co., Chapt 9, (1997).

[EE92] S. Shoji and M. Esashi., *Micro-flow Devices and Systems*, Journal of Micro-mechanics and Micro-engineering (1992).

[S94] Sinden, R. *DNA Structure and Function*, Academic Press, (1994).

[S88] Smith, L.M. *Automated Synthesis and Sequence Analysis of Biological Macromolecules*, Anal. Chem., **60**, 381A–390A, (1988).

[SS95] Smith, W., and A. Schweitzer, *DNA Computers in Vitro and Vivo*, NEC Research Inst. Tech Report 95-057-3-0058-3. (1995).

[StM97] Stefan, G., and M. Malita, *The Splicing Mechanism and the Connex Memory*, ICEC'97 Special Session on DNA Based Computation, Indiana, April, 1997.

[S71] H.S. Stone *Parallel Processing with the perfect shuffle*, IEEE Trans. on Computers, **C-20**:2, 153-161 (1971).

[Suy98] A. Suyama, *DNA chips - Integrated Chemical Circuits for DNA Diagnosis and DNA computers*, to appear, (1998).

[U84] J. Ullman, *Computational Aspects of VLSI*, Computer Science Press, (1984), Chapter 6.

[VSJMWR92] E. M. J. Verpoorte, van der Schoot, B. H., S. Jeanneret , A. Manz, H. M. Widmer, and de Rooij, N. F., *Three-Dimensional Micro-flow Manifolds for Miniaturized Chemical Analysis Systems*, Journal of Micro-mechanics and Micro-engineering, (1992).

[W61] H. Wang, *Proving Theorems by Pattern Recognition*, Bell System Technical Journal, **40**, 1–141, (1961).

[WQF+98] Wang, L., Q. Liu, A. Frutos, S. Gillmor, A. Thiel, T. Strother, A. Condon, R. Corn, M. Lagally, L. Smith, *Surface-based DNA computing operations: DESTROY and READOUT"*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[WGWZ92] Watson, J., M. Gilman, J. Witkowski, M. Zoller, *Recombinant DNA (2nd ed.)*, Scientific American Books, W.H. Freeman and Co., (1992).

[WHR87] Watson, J., N. Hoplins, J. Roberts, et. al., *Molecular Biology of the Gene*, Benjamin/Cummings, Menlo Park, CA, (1987).

[W97] Wetmur, J. G. *Physical Chemistry of Nucleic Acid Hybridize*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[WW95] Williams, R.M., and David H. Wood, *Computational algebras for RNA processes*, Gene-Finding and Gene Structure Prediction Workshop, Unrefereed poster presentation, (1995).

[WW96] Williams, R.M., and David H. Wood, *Exascale Computer Algebra Problems Interconnect with Molecular Reactions and Complexity Theory*, The 2nd Annual Workshop on DNA Based Computers, American Mathematical Society, June, 1996.

[W95] Winfree, E. *Complexity of Restricted and Unrestricted Models of Molecular Computation*, California Institute of Technology technical report May, 1995. Also Princeton DIMACS Technical Report workshop on DNA-based computers, April 4, 1995.

[W96] Winfree, E. *On the computational power of DNA annealing and ligation*, DNA based computers, Lipton, R.J. and Baum, E.B. eds., Am. Math. Soc., Providence, RI, (1996).

[Win98a] Winfree, E., *Simulations of computing by self-assembly*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[Win98b] Winfree, E., *Whiplash PCR for O(1) computing*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[WLW+98] Erik Winfree, Furong Liu, Lisa A. Wenzler, Nadrian C. Seeman, *Design and Self-Assembly of Two Dimensional DNA Crystals*, Nature 394: 539–544, 1998. (1998).

[WYS96] Winfree, E., X. Yang, N.C. Seeman, *Universal Computation via Self-assembly of DNA: Some Theory and Experiments*, 2nd Annual DIMACS Meeting on DNA Based Computers, Princeton, June, 1996.

[YK97a] Yokomori, T., and S. Kobayashi, *On the Power of Circular Splicing Systems and DNA Computability*, ICEC'97 Special Session on DNA Based Computation, Indiana, April, 1997.

[YK97b] Yokomori, T., and S. Kobayashi, *DNA-EC: A Model of DNA-Computing Based on Equality Checking*, 3rd DIMACS Meeting on DNA Based Computers, Univ. of Penns., (June, 1997).

[Woo98] Wood, D. H., *Applying error correcting codes to DNA computing*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[YAT+98] Yoshinobu, T., Y. Aoi, K. Tanizawa, Hiroshi Iwasaki, *Ligation errors in DNA computing*, 4th Int. Meeting on DNA-Based Computing, Baltimore, Penns., (June, 1998).

[ZS92] Zhang, Y., and N.C. Seeman, *A Solid-Support Methodology for the Construction of Geometrical Objects from DNA*, J. Am. Chem. Soc., **114**, 2656–2663, (1992).

[ZS94] Zhang, Y., and N.C. Seeman, *The Construction of a DNA Truncated Octahedron*, J. Am. Chem. Soc., **116**, 1661–1669, (1994).