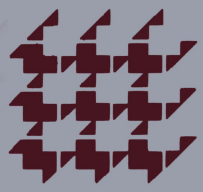

DIMACS

Series in Discrete Mathematics and Theoretical Computer Science

Volume 54

DNA Based Computers V

DIMACS Workshop
DNA Based Computers V
June 14–15, 1999
Massachusetts Institute of Technology

Erik Winfree
David K. Gifford
Editors

American Mathematical Society

Selected Titles in This Series

54 **Erik Winfree and David K. Gifford, Editors,** DNA Based Computers V

53 **Nathaniel Dean, D. Frank Hsu, and R. Ravi, Editors,** Robust Communication Networks: Interconnection and Survivability

52 **Sanguthevar Rajasekaran, Panos Pardalos, and D. Frank Hsu, Editors,** Mobile Networks and Computing

51 **Pierre Hansen, Patrick Fowler, and Maolin Zheng, Editors,** Discrete Mathematical Chemistry

50 **James M. Abello and Jeffrey Scott Vitter, Editors,** External Memory Algorithms

49 **Ronald L. Graham, Jan Kratochvíl, Jaroslav Nešetřil, and Fred S. Roberts, Editors,** Contemporary Trends in Discrete Mathematics

48 **Harvey Rubin and David Harlan Wood, Editors,** DNA Based Computers III

47 **Martin Farach-Colton, Fred S. Roberts, Martin Vingron, and Michael Waterman, Editors,** Mathematical Support for Molecular Biology

46 **Peng-Jun Wan, Ding-Zhu Du, and Panos M. Pardalos, Editors,** Multichannel Optical Networks: Theory and Practice

45 **Marios Mavronicolas, Michael Merritt, and Nir Shavit, Editors,** Networks in Distributed Computing

44 **Laura F. Landweber and Eric B. Baum, Editors,** DNA Based Computers II

43 **Panos Pardalos, Sanguthevar Rajasekaran, and José Rolim, Editors,** Randomization Methods in Algorithm Design

42 **Ding-Zhu Du and Frank K. Hwang, Editors,** Advances in Switching Networks

41 **David Aldous and James Propp, Editors,** Microsurveys in Discrete Probability

40 **Panos M. Pardalos and Dingzhu Du, Editors,** Network Design: Connectivity and Facilities Location

39 **Paul W. Beame and Samuel R Buss, Editors,** Proof Complexity and Feasible Arithmetics

38 **Rebecca N. Wright and Peter G. Neumann, Editors,** Network Threats

37 **Boris Mirkin, F. R. McMorris, Fred S. Roberts, and Andrey Rzhetsky, Editors,** Mathematical Hierarchies and Biology

36 **Joseph G. Rosenstein, Deborah S. Franzblau, and Fred S. Roberts, Editors,** Discrete Mathematics in the Schools

35 **Dingzhu Du, Jun Gu, and Panos M. Pardalos, Editors,** Satisfiability Problem: Theory and Applications

34 **Nathaniel Dean, Editor,** African Americans in Mathematics

33 **Ravi B. Boppana and James F. Lynch, Editors,** Logic and Random Structures

32 **Jean-Charles Grégoire, Gerard J. Holzmann, and Doron A. Peled, Editors,** The SPIN Verification System

31 **Neil Immerman and Phokion G. Kolaitis, Editors,** Descriptive Complexity and Finite Models

30 **Sandeep N. Bhatt, Editor,** Parallel Algorithms: Third DIMACS Implementation Challenge

29 **Doron A. Peled, Vaughan R. Pratt, and Gerard J. Holzmann, Editors,** Partial Order Methods in Verification

28 **Larry Finkelstein and William M. Kantor, Editors,** Groups and Computation II

27 **Richard J. Lipton and Eric B. Baum, Editors,** DNA Based Computers

26 **David S. Johnson and Michael A. Trick, Editors,** Cliques, Coloring, and Satisfiability: Second DIMACS Implementation Challenge

For a complete list of titles in this series, visit the AMS Bookstore at **www.ams.org/bookstore/**.

This page intentionally left blank

https://doi.org/10.1090/dimacs/054

DIMACS

Series in Discrete Mathematics
and Theoretical Computer Science

Volume 54

DNA Based Computers V

DIMACS Workshop
DNA Based Computers V
June 14–15, 1999
Massachusetts Institute of Technology

Erik Winfree
David K. Gifford
Editors

NSF Science and Technology Center
in Discrete Mathematics and Theoretical Computer Science
A consortium of Rutgers University, Princeton University,
AT&T Labs–Research, Bell Labs (Lucent Technologies),
Telcordia Technologies, and NEC Research Institute

American Mathematical Society

This DIMACS volume contains papers from a DIMACS workshop entitled DNA Based Computers V, held at the Massachusetts Institute of Technology, Cambridge, MA, June 14–15, 1999.

2000 *Mathematics Subject Classification.* Primary 68–06, 68Q05, 68Q10, 68M99, 92–06, 92B05, 92B20, 92C05.

Library of Congress Cataloging-in-Publication Data

DIMACS Workshop on DNA Based Computers (5th : 1999 : Massachusetts Institute of Technology)

DNA Based Computers V : DIMACS Workshop DNA Based Computers, June 14–15, 1999, Massachusetts Institute of Technology / Erik Winfree, David K. Gifford, editors.

p. cm. — (DIMACS series in discrete mathematics and theoretical computer science ; v. 54)

Includes bibliographical references.

ISBN 0-8218-2053-2 (alk. paper)

1. Molecular computers—Congresses. I. Winfree, Erik, 1969– II. Gifford, David K., 1954– III. Series.

QA76.887.D5 1999

511.3—dc21 00-040621

Contents

Foreword

The Fifth International Meeting on DNA Based Computers was held at MIT on June 14–15, 1999. We would like to express our appreciation to Erik Winfree and David Gifford for their efforts to organize and plan this successful conference.

The workshop was part of the Special Focus on DNA Computing. We extend our thanks to Laura Landweber and Richard Lipton for their work as special focus organizers.

The conference brought together theoreticians and practitioners working on DNA computing from a variety of perspectives, with an emphasis on DNA computing based on combinatorial search and on DNA computing based on finite-state machines.

DIMACS gratefully acknowledges the generous support that makes these programs possible. The National Science Foundation, through its Science and Technology Centers program, the New Jersey Commission on Science and Technology, and DIMACS's partners at Rutgers, Princeton, AT&T Labs-Research, Bell Labs, NEC Research Institute, and Telcordia Technologies generously supported the special focus.

Fred S. Roberts
Director

Robert Sedgewick
Co-Director for Princeton

This page intentionally left blank

Introduction

The Fifth International Meeting on DNA Based Computers was held at MIT June 14-15, 1999 and presented papers that explored two distinct styles of DNA computing:

1. DNA computing based on combinatorial search, where randomly created DNA strands are used to encode potential solutions to a problem, and constraints induced by the problem are used to identify DNA strands that are solution witnesses.
2. DNA computing based on finite-state machines, where the state of a computation is encoded in DNA, which controls the biochemical steps that advance the DNA-based machine from state to state.

Early results in DNA computing focused on idealized combinatorial search. The first result by Leonard Adleman in *Science* titled "Molecular Computation of Solutions to Combinatorial Problems" showed how to solve a small instance of the directed hamiltonian path problem. Pure combinatorial search proposes to use exponential amounts of DNA instead of the exponential amount of time used by traditional approaches. Lipton, Boneh, and colleagues presented molecular algorithms for solving formula and circuit satisfiability problems, and several other groups showed that universal Turing machines can be simulated in parallel, with a single DNA molecule encoding the tape and head state of each machine. However, despite some very real experimental advances, laboratory demonstrations of DNA computing are still firmly in Toyland. The question on everyone's mind during the meeting was, "Can we get out of Toyland?"

The meeting was organized into four half-day sessions: Satisfying Experiments, Design and Simulation, Self-Assembly and Nanomachines, and Theoretical Perspectives. After each session, a panel discussion was held to bring into the open questions about where the research is going and what needs to be done next.

The first session opened with three back-to-back talks each describing the successful solution of a small Formula Satisfiability problem using a different technique. The paper by **Dirk Faulhammer, Anthony Cukras, Richard Lipton, and Laura Landweber** describes their work using RNA. The Princeton group's work is notable not only because it is the largest SAT problem yet attempted, with 9 input variables, but perhaps more importantly because they have made a considerable effort to characterize the error rates and error mechanisms in individual reactions.

Hiroshi Yoshida and Akira Suyama describe their work developing methods for solving 3-SAT by breadth-first search. The University of Tokyo researchers' goal is to find more efficient algorithms than the 'classical' brute-force approach

wherein all possible solutions are generated and tested. In the breadth-first algorithm, solutions are constructed recursively, first finding all 1-variable solutions, then expanding those to all 2-variable solutions, and so forth. Empirical trials of the algorithm suggest that the number of different DNA strands required for 'typical' problems scales as $2^{\frac{n}{2}}$ instead of 2^n – a dramatic improvement (but not uncommon in computer science) that leads the authors to think that 100-variable problems might be solved by this method. It remains an open question whether the most difficult 3-SAT problems will succumb to the breadth-first approach, and whether breadth-first algorithms for formula-SAT or circuit-SAT can be devised. The algorithm presented here is particularly elegant, with all key steps experimentally prototyped on a 4-variable formula; importantly, error rates were also measured.

The talk by **Liman Wang** described the Wisconsin group's recent success doing DNA computing on surfaces. Their effort works to bring to DNA computing the lesson of automated DNA synthesis: that it is easy to subject a population of DNA strands to a sequence of chemical reagents when the DNA is attached to a solid support. They have now developed the technology to fully implement their algorithm on a 4-variable 3-SAT problem. An ancillary, but also important, accomplishment of the Wisconsin group is that, at least for small problems, DNA chips can be used to read-out the entire set of DNA sequences at any point in the algorithm, allowing them to measure the sequence-dependence of error rates – knowledge that will be critical for scaling up all DNA computing approaches. Unfortunately, their paper will not appear in this volume.

The discussion panel brought up many tough questions about the feasibility of solving significant SAT problems with DNA. A critical question, if SAT is to remain a major target for DNA based computers, is whether there are any problems in the size range DNA based computers might be able to solve that would pose any real difficulty for electronic computers. It was pointed out that modern heuristic algorithms such as Walk-SAT can routinely solve "typical" thousand-variable 3-SAT problems as well as the hardest randomly-generated 250-variable problems, and there are other 3-SAT algorithms for which $O(1.61^n)$ bounds have been proved. But are there any hard 100 variable problems, of the scale that DNA computers can potentially handle?

Going one step beyond Yoshida and Suyama on the road from rigorous algorithms to heuristics, **Junghuei Chen, Eugene Antipov, Bertrand Lemieux, Walter Cedeño, and David Wood** described preliminary results applying the logic of evolutionary computation to DNA-based computation. Although their experimental protocol is designed to solve a trivially simple problem, it is one that has been the subject of many theoretical studies of genetic algorithm performance. It will be interesting, in the future, to experimentally evaluate the effectiveness of genetic algorithms on large DNA population. It would be a welcome turn of events if the synthesis of evolutionary computation and DNA based computers also leads to better ways to evaluate the effectiveness of traditional in-vitro evolution and finds methods to improve them.

As hinted above, understanding and controlling error rates are central to solving large problems with DNA based computers. If at each step, a strand has even a 10% chance of being misprocessed or accidentally destroyed, after just 100 steps only 1 in 40,000 strands will have been processed correctly. Addressing this issue, **Kevin Chen and Erik Winfree** analyzed error-correcting algorithms for both separation errors and for strand loss errors, presuming that sequence independence

of error rates can be achieved by biochemical means. However, errors are not the only potential bottleneck to scaling up DNA based computations to significant problems. The limiting factors could equally well be time if many steps are required, or strand length, or test tube complexity. It behooves us to figure out which are the critical bottlenecks. New implementations that allow more computation to occur in a single test tube, such as the use of liposomes to compartmentalize reactions as proposed by **Brian Bloom and Carter Bancroft** or the self-assembly approaches mentioned below, may make applications to larger problem instances feasible. Overall, things look pretty good. I will even venture to bet that a 20-variable SAT problem will be solved within the year. However, we are still very far from doing solid engineering, where we built systems with reliable performance by using well-characterized components; we do not yet have the intellectual framework for characterizing and predicting the behavior of molecular systems to the required level of accuracy.

To make matters more complicated, the set of available molecular tools is ever increasing. In a captivating invited talk, **Eric Kool** described three molecular technologies he has been developing. Closest to the atomic scale is a new pair of artificial DNA "bases" that can be incorporated by DNA polymerases. They are remarkable because one is essentially nothing (an abasic site) and the side group on the other is a large four-benzene-ring structure, pyrene, and thus specificity is not mediated by hydrogen bonds. At a slightly larger scale are activated DNA oligonucleotides, with modified $3'$ and $5'$ ends, that can easily be ligated without the need for enzymes or special reagents. His last example made novel use of an enzyme by asking RNA polymerase to transcribe from small circular oligonucleotides – somewhat like teaching an elephant to balance on a circus ball. Indeed, in this reaction, "rolling circle transcription," the polymerase will go around and around the template, producting long transcripts – which, if they contain the appropriate ribozyme motif, in turn can self-cleave to produce a well-defined product.

The second session on Design and Simulation focussed on how to design DNA computers within the limits set by physics. The paper by **Allen Mills, Bernard Yurke, and Phil Platzman** describes how parallel DNA hybridization reactions could implement vector algebra operations, including neural networks. After analysing the cycle time and statistical limits for this form of computation, he concluded that vector spaces as large as 10^7 dimensions are conceivable. An unsolved problem is how to find a set of 10^7 oligonucleotides with no or very little cross-hybridization. **Amit Marathe and Anne Condon** addressed this problem by defining several criteria related to cross-hybridization, and finding bounds on the size of sets of strings that satisfy the criteria. **Max Garzon, Russell Deaton, John Rose, and Donald Franceschetti** discuss a different set of criteria for the same problem, which have been implemented in the software package `Edna`. **Masahito Yamamoto et al.** have their own criteria for designing non-interacting DNA sequences; their work is notable for having an experimental investigation of the sequences they designed. **Alexander Hartemink, Tarjei Mikkelsen, and David Gifford** describe modular software for simulating biological reactions; submodules include prediction of hybridization and the action of enzymes such as polymerase and ligase. The package has been used to predict the products of the PCR-based unary counter that their laboratory has been investigating experimentally. Such a tool can be useful not only for prediction, but also for identifying hypotheses for unexpected experimental results.

Panel discussion identified several challenges for simulation and design software. The first is how to assess validity and scope – that is, how to compare the simulation results to real experiments and understand the limits for accurate prediction of new experiments. There was discussion of what the fundamental data structures should be for such as simulation; should modelling be done at an abstract logical level, at a quantitative thermodynamic level, at the atomic or quantum level, or do all these levels of abstractions need to be combined to accurately model molecular systems? As a practical matter, there was general agreement that algorithms and code should be made available on the Web, to promote communication between labs; it is a bit frustrating that every lab has their own DNA design criteria, and no one can convincingly argue that one is unambiguously better than another.

The third session on Self-Assembly and Nanomachines explored some alternative approaches to and alternative uses for molecular computation. **Thomas LaBean, Erik Winfree, and John Reif** report experimental progress toward implementing the String Tile model, which holds promise using a single self-assembly reaction to construct a molecular lookup table for addition. **Michail Lagoudakis and Thomas LaBean** give a theoretical example of performing more complex computations by 2D self-assembly: they show how to construct a set of tiles that solve the satisfiability problem in a single self-assembly reaction. **Takashi Yokomori** proposes a new model of computation that involves a particularly simple experimental reaction: the self-assembly of linear chains of pre-formed components, followed by melting and re-hybridization into a low-energy structure. Remarkably, this model is Turing universal.

Andrew Turberfield, Bernard Yurke, and Allen Mills describe a series of fiendishly clever, yet remarkably simple, experiments where DNA oligonucleotides are used as catalysts to control the hybridization of other DNA molecules. This careful and quantitative work is sure to serve as the groundwork for more elaborate DNA mechanisms. In fact, the group have already constructed their first "machine", a jointed DNA molecule that can be opened and closed like tweezers. A paper by **Michael Robertson, Jay Hesselberth, Colin Cox, and Andrew Ellington** describes the group's work using *in vitro* evolution and rational modification to discover ribozymes that can ligate DNA under control of a complex switch that can depend on, so far, up to three inputs, such as a DNA oligonucleotide, adenosine, and theophyline. Combining these switches into interacting molecular systems will be a fascinating endeavour.

It is interesting to speculate on where this work is going. DNA and RNA can perform logical computations during self-assembly; it can catalyze and perform mechanical work on other DNA molecules; it can undergo conformational switching in response to effector molecules; it can catalyze chemical reactions; are these not sufficient building blocks for constructing arbitrarily sophisticated biochemical systems and molecular devices? Might we be creating a "DNA world" as a stage in molecular engineering, regardless of whether such a stage existed in the evolution of life on Earth? These were all topics of the panel discussion.

What theoretical constructs will be necessary for molecular computation and complex biochemical systems? What theoretical goals can help use guide experimental investigations into promising areas? This was the topic of the fourth and final session, Theoretical Perspectives. The paper by **Andrzej Ehrenfeucht, Hendrik Jan Hoogeboom, Grzegorz Rozenberg, and Nikè van Vugt** gives an elegant mathematical analysis of sets generated by simultaneous enforcing and

forbidding conditions. An advantage of this formalism is that it does not depend explicitly on the order of elementary events – a key property for describing molecular systems where reactions occur asynchronously. **Lila Kari and Laura Landweber** ask whether concepts from computer science can help us understand biological phenomena, such as the mysterious gene unscrambling found in single-cell ciliates. **Gheorghe Păun and Takashi Yokomori** present new results for a family of models inspired after the subcellular compartments found in Eukaryotic cells. Going ever smaller, **Ehud Shapiro** presented his model of computation in which a hypothetical holoenzyme – like the ribosome but tailored for computation – acts on a DNA-like polymer to simulate a Turing Machine, possibly producing molecules as output. His paper will appear elsewhere. The paper by **Ashish Gehani, Thomas Labean, and John Reif** analyses a new and unexpected direction for DNA-based cryptography: rather than try to break electronic cryptosystems, use DNA to encrypt and transmit secret messages. It seems there is no shortage of surprises in the molecular world.

During the panel discussion it was noted that in fields such as computer science and quantum physics, theory preceeds experiment, which only goes so far as to find a way to obtain what the theory predicts; whereas in fields such as biology and solid-state physics, experiment preceeds theory, which only manages to put the pieces together and explain them in a more understandable form. So far, DNA based computing has been driven by theory. But it is quite possible, as unexpected molecular computing components are discovered, as continually occurs in Ellington's lab, that the reverse trend will become dominant.

So, the future of DNA computing looks bright. But don't reach for your sunglasses. It is still before the dawn, and in the dark shadows I just stubbed my toe on something.

Erik Winfree

David Gifford

Organizers & Editors

This page intentionally left blank

ISBN 0-8218-2053-2

DIMACS/54