

Sudheer Sahu

Address: Department of Computer Science, Box 90129, Duke University, Durham, NC 27705-0129
Phone: 919-660-6512 (work)
919-452-5843 (cell)
Email: sudheer@cs.duke.edu
Homepage: <http://www.cs.duke.edu/~sudheer>

Research Interests

Design and Analysis of Algorithms, Stochastic Modeling, Randomized Algorithms, Self-Assembly, DNA computing, Nanorobotics, Complexity Theory, Graph Theory, Fault Tolerance.

Education

2002-present: **PhD, Computer Science, Duke University** (Expected 2007)
Dissertation: *Theory and Experiments in Self-Assembly and Nanorobotics*
Advisor: John H. Reif
Committee: John Board, Alex Hartemink, Thom LaBean, Kamesh Munagla, Xiaobai Sun.
Fault-tolerance in algorithmic self-assembly; New and powerful models in self-assembly; Algorithmic and complexity theoretic problems in self-assembly; Stochastic modeling of DNA based nanorobotical devices; Design and implementation of novel DNA nanorobotical devices

1998-2002: **Bachelor of Technology**
Computer Science and Engineering, **IIT (Indian Institute of Technology)**, New Delhi.

Research Experience

May'03-present **Research Assistant**, Department of Computer Science, Duke University
Fault-tolerance in self-assembly; Design of new and powerful models in Self-Assembly; Stochastic modeling of DNA based nanorobotical systems; Algorithmic, graph-theoretic and complexity-theoretic problems in self-assembly; Stochastic analysis of reversible self-assembly; Design and experimental implementation of DNA nanorobotical devices; Experimental implementations in DNA based self-assembly.

May-July'01 **Research Intern**, Max Planck Institut fur Informatik, Saarbrucken, Germany
Single-source shortest-paths on arbitrary directed graphs in linear average-case time
(Mentor: Dr. Ulrich Meyer)
The project involved implementing a *SSSP* (*single source shortest path*) algorithm for arbitrary directed graphs with random edge weights uniformly distributed in $(0,1]$. The algorithm based upon *adaptive bucket splitting* and *label-correcting approach* ran in $O(n + m)$ time with high probability.

Teaching Experience

- CPS150: Introduction to Numerical Analysis. Teaching Assistant, Duke University (Fall 07).
- CPS230: Design and Analysis of Algorithms. Teaching Assistant, Duke University (Fall 06).
- CPS237: Randomized Algorithms. Teaching Assistant, Duke University (Spring 06).
- CPS230: Design and Analysis of Algorithms. Teaching Assistant, Duke University (Fall04).
- CPS150: Introduction to Numerical Analysis. Teaching Assistant, Duke University (Fall03).
- CPS140: Mathematical Foundations of Computer Science. Teaching Assistant, Duke University. (Spring03)
- CPS240: Computational Complexity. *Tiling Theory: Tile Complexity and Time Complexity in Irreversible Model*, **Guest Lecturer**, Duke University (Spring 07).

Industrial Experience

- June-July'02 **Intern**, Intersolutions Pvt Ltd, Noida, New Delhi
Design and development of Java, J2EE based complex workflow/rule based applications with Oracle database
- May-July'00 **Student Intern**, Training and Placement Department, IIT Delhi
Web-based automated application system for campus recruitment in IIT Delhi
 (Supervisor: Dr. Y.S.Goel)
 The project involved *cgi scripting* in perl and *postgresql* database manager.

Publications

1. John H. Reif, Sudheer Sahu, "Autonomous Programmable DNA Nanorobotical Devices using DNAzymes", *Fourth Conference on Foundations of nanoscience: self-assembled architectures and devices (FNANO07), Snowbird, April 18-21, 2007.*

We designed a *DNAzyme calculator*, a finite state automata device (*DNAzyme FSA*) that executes finite state transitions using DNAzymes, its extension to probabilistic and non-deterministic automata, their applications as *DNAzyme router*, and *DNAzyme porter*. In addition we present *DNAzyme doctor* that can provide transduction of nucleic acid expression: it can be programmed to respond to the under-expression or over-expression of various strands of RNA, with a response by release of an RNA. For the *stochastic modelling* of the these devices, the activity of a DNAzyme is modelled as a *reversible markov chain* and the strand displacement process as a *biased random walk*.

2. Urmi Majumder, Sudheer Sahu, and John H. Reif, "Reversible self-assembly of squares as a rapidly mixing markov chain", *Fourth Conference on Foundations of nanoscience: self-assembled architectures and devices (FNANO07), Snowbird, April 18-21, 2007.*

This paper presents *stochastic modelling* of *reversible self-assembly* process that includes the determination of equilibrium concentrations and *rate of convergence* to equilibrium. We prove that an assembly in chemical equilibrium implies that the corresponding *markov chain* has reached its *ergodic limit*. Based on the convergence rate analysis for the self-assembly of linear systems and assembly of $n \times n$ square, we conclude that the corresponding markov chains are *rapidly mixing*. In fact, the tile system continues to converge to equilibrium exponentially fast even when we allow errors in the system. We extend the analysis to general shapes in two-dimensions and three-dimensional assembly of a $n \times n \times n$ cube.

3. Sudheer Sahu, Bei Wang, Peng Yin, John H. Reif, "Framework for Modeling DNA based Molecular Systems", *Twelfth International Meeting on DNA Based Computers (DNA12), Seoul, Korea, June 5-9, 2006. Lecture Notes in Computer Science, Springer-Verlag, New York, (2006).*

In this paper, we present a framework for a *discrete event simulator* for simulating the DNA based nanorobotical systems. The conformational changes of long chain DNA molecules are modelled using *metropolis algorithm*. The molecular motion is modelled as a *stochastic process (brownian motion)*, and *collision detection* methods are employed to detect molecular collisions. Various chemical reactions like hybridization/dehybridization are modelled as *markov chains* and strand displacement is modelled as a *biased random walk*.

4. Sudheer Sahu, John Reif, "Capabilities and Limits of Compact Error Resilience Methods for Algorithmic Self-Assembly in Two and Three Dimensions", *Twelfth International Meeting on DNA Based Computers (DNA12), Seoul, Korea, June 5-9, 2006. Lecture Notes in Computer Science, Springer-Verlag, New York, (2006).*

This paper presents a comprehensive theory of compact methods of fault tolerance in algorithmic self-assembly: an extension to our previous work on compact error resilient algorithmic self-assemblies. We present a compact error resilient scheme that reduces error from ϵ to ϵ^2 for algorithmic assemblies with any arbitrary Boolean functions. Also, we characterize a restricted class of input-sensitive Boolean functions only within which error reduction from ϵ to ϵ^3 is possible using compact error resiliency, and proposed a scheme for the same. Further we prove that reduction of errors from ϵ to ϵ^4 is not possible for any algorithmic assembly using compact error resilient methods. In addition to this, similar fault tolerance theory for three dimensional algorithmic self-assemblies is developed.

5. Peng Yin, Sudheer Sahu, Rizal Hariadi, Harry M. T. Choi, Sung Ha Park, Bethany Walters, Thomas H. LaBean, John H. Reif, "On Constructing Tile-less DNA Ribbons and Tubes", *Twelfth International Meeting on DNA Based Computers (DNA12), Seoul, Korea, June 5-9, 2006.*

This paper describes a new *tile-less* approach to construct DNA lattice structures, in which the intermediate construct of tiles is bypassed. Single stranded DNA is used to directly form the intended DNA lattice. The tile-less approach, intended to serve as a complementary approach to the current dominant paradigm of tile-based approach, can often offer the following desirable properties: conceptual and structural simplicity, finer-grained programmability/higher information density, and higher thermal stability. We describe a general scheme that has been proved successful for the construction of extended DNA ribbons with programmable width and the construction of long DNA tubes.

6. Urmi Majumder, Sudheer Sahu, Thom LaBean, John H. Reif, "Design and Simulation of Self-Repairing DNA Lattices", *Twelfth International Meeting on DNA Based Computers (DNA12)*, Seoul, Korea, June 5-9, 2006. *Lecture Notes in Computer Science*, Springer-Verlag, New York, (2006).

The paper describes a special class of DNA tiling designs called *reversible tiling* designs which when carefully designed can provide inherent self-repairing capabilities to patterned DNA lattices so long crystal growth/regrowth happens with respect to at least two adjacent binding sites. We discuss the design, simulation and a preliminary DNA implementation of a particular instance of carefully designed reversible tiling which we call *Reversible XOR*. We further note that although in theory one can transform any irreversible computational DNA tile set to its reversible counterpart and hence improve the self-repairability of the computational lattice.

7. John H. Reif, Sudheer Sahu, Peng Yin, "The Complexity of Graph Self-Assembly in Accretive Systems and Self-Destructible Systems", *Eleventh International Meeting on DNA Based Computers (DNA11)*, Ontario, Canada, June 6-9, 2005. *Lecture Notes in Computer Science*, Springer-Verlag, New York, (2005).

Complexity theoretic analysis of *Graph Assembly Problem (GAP)*: NP-completeness of Accretive GAP, #P-completeness of stochastic AGAP and P-SPACE completeness of Self-destructible GAP.

8. Peng Yin, Sudheer Sahu, John H. Reif, "Autonomous DNA Cellular Automata", *Eleventh International Meeting on DNA Based Computers (DNA11)*, Ontario, Canada, June 6-9, 2005. *Lecture Notes in Computer Science*, Springer-Verlag, New York, (2005).

In this paper, we describe the design of an Autonomous DNA Cellular Automaton (ADCA), which can perform *parallel universal computation* by mimicking a one-dimensional (1D) universal cellular automaton. The states and transition-rules are encoded in the various DNA sequences used in the design. The key technical innovation is "reaction waves": a mechanism to *synchronize* and *pipeline* events that are otherwise independent.

9. Sudheer Sahu, Peng Yin, John H. Reif, "A Self-Assembly Model of DNA Tiles with Time Dependent Glue Strength", *Eleventh International Meeting on DNA Based Computers (DNA11)*, Ontario, Canada, June 6-9, 2005. *Lecture Notes in Computer Science*, Springer-Verlag, New York, (2005).

We propose a theoretical self-assembly model known as *Time Dependent Glue (TDG) model*, in which the glue strength between two juxtaposed tiles is a function of the time they have been in neighboring positions. We model the processes of catalysis and self-replication under TDG, and study the *tile-complexity* for assembling shapes in our model and show that a thin rectangle of size $k \times N$ can be assembled using $O(\frac{\log N}{\log \log N})$ types of tiles, for any constant $k > 0$.

10. John H. Reif, Sudheer Sahu, Peng Yin, "Compact Error-Resilient Computational DNA Tiling Assemblies", *Tenth International Meeting on DNA Based Computers (DNA10)*, Milano, Italy, June 7-10, 2004. *Lecture Notes in Computer Science*, Springer-Verlag, New York, (2004).

This paper describes the designs for compact methods of *fault-tolerance* in *algorithmic self-assembly* known as *compact error resilient schemes*. The scheme, based on 2-way *overlay redundant computations*, reduces the error probability from ϵ to ϵ^2 for algorithmic assemblies having at least one Boolean function as *XOR*. The error probability is further improved to ϵ^3 using 3-way redundant computations for a restricted class of Boolean functions.

11. Peng Yin, Andrew J. Turberfield, Sudheer Sahu, John H. Reif, "Design of an Autonomous DNA Nanomechanical Device Capable of Universal Computation and Universal Translational Motion", *Tenth International Meeting on DNA Based Computers (DNA10)*, Milano, Italy, June 7-10, 2004. *Lecture Notes in Computer Science*, Springer-Verlag, New York, (2004).

This paper presents a design of a nanomechanical DNA device that autonomously mimics the operation of a *2-state 5-color universal Turing machine*. The states and transition-rules of turing machine are encoded in the various DNA sequences used in the design.

12. John H. Reif, Thomas LaBean, Sudheer Sahu, Hao Yan, Peng Yin, "Design, Simulation, and Experimental Demonstration of Self-Assembled DNA Nanostructures and DNA Motors", *Computational Modeling and Simulation of Materials (CIMTEC) Conference, Acireale, Sicily, Italy, May 29-June 4, 2004*.

We present an overview of the theoretical models, algorithms and our softwares for simulation and design of DNA tiling assemblies and DNA nano-mechanical devices. We also discuss our laboratory demonstrations of DNA lattices and motors, including those using the designs aided by our software.

In Preparation

- John H. Reif, Sudheer Sahu, "Optimal Algorithms for Oblivious Group Testing Problems". In preparation.

We present a randomized algorithm for oblivious group testing that detect $d=O(n)$ defective items from a set of n objects in $O(d \log(n/d))$ oblivious tests, where a test can determine the existence of any defective item in that test-sample. We prove the matching lower bound on number of tests, hence showing the optimality.

- Sudheer Sahu, John H. Reif, "Approximate Pattern Matching for Entropy Bounded Text". In preparation.

We present an algorithm that runs in $O(n \log m/pm)$ expected time where H is the entropy of the text and $p = 1 - (1 - H^2)^{\frac{H}{1+H}}$ based on preprocessing of the text.

- Peng Yin, Bo Guo, Christina Belmore, Sudheer Sahu, Will Palmeri, Erik Winfree, Thomas H. LaBean, John H. Reif, "TileSoft: Sequence Optimization Software for Designing DNA Secondary Structures". In Preparation.

A crucial computational problem in constructing DNA objects is the design of DNA sequences that can correctly assemble into desired DNA secondary structures. TileSoft described here deliver the following features: 1) Its graphical user interface renders the molecular architect the ability to define DNA secondary structure and accompanying designing constraints directly on the interface as well as the ability to view the optimized sequence information pictorially. 2) Its fully automatic optimization module relieves the user of the drudgery of manually dictating the sequence selection process, and its evolutionary algorithm produces satisfactory results efficiently. 3) Its graphical user interface and its optimization module are smoothly integrated from user's perspective, while they are at the same time well separated in terms of software architecture, making each amenable to future improvements without negatively affecting the other.

Technical Reports

1. Sudheer Sahu, Deepak Ajwani, "Cache-Efficient Fast Fourier Transform", *B.Tech Project, IIT Delhi, 2002*.

Initially, an *I/O model FFT* algorithm was designed that performed each stage of *butterfly computation* on small subsets of input interleaved with matrix-transposition. This I/O model algorithm was used to develop a *cache-efficient FFT* algorithm by application of *emulation theorem*. In terms of the number of cache hits *cache-efficient FFT* outperformed *FFTW* (fastest fourier transform in the west), while the running time of cache-efficient FFT was found comparable to that of FFTW.

2. Sudheer Sahu, "Single-source shortest-paths on arbitrary directed graphs in linear average-case time", *Max Planck Institut fur Informatik, July 2001*.

The project involved implementing a *SSSP(single source shortest path)* algorithm for arbitrary directed graphs with random edge weights uniformly distributed in $(0,1]$. The algorithm based upon *adaptive bucket splitting* and *label-correcting approach* ran in $O(n + m)$ time with high probability.

3. Sudheer Sahu, Anshul Sawant, "Terrain Visualisation for 3D Geometric Terrains", *Mini-Project, IIT Delhi, 2001*.

Delauney triangulation of an input terrain is used to extract maps and silhouettes, which are then stored in *horizon trees*. A variant of *sweep-line algorithm* is used to clip the map and silhouette edges, and the *monotone chains* for a viewer-position are computed to display of the terrain. The algorithm is able to handle even those viewer-movements that break the monotonicity of the monotone chains.

Selected Course Projects

1. “Clustering Error-Prone Gene Expression Data”. (*Computational Functional Genomics*, Spring 03)
Optimized the *PCC and CAST algorithms* (given by Ben-Dor) for clustering erroneous gene-expression data.
2. “Genome Rearrangement with Gene Families”. (*Algorithms in Computational Biology*, Fall 02)
Improved the Sankoff’s algorithm for finding the *exemplar breakpoint distance* between two genomes by pruning the search tree and simultaneously optimizing the search procedure at the nodes.
3. “Accurate Static Branch Prediction using Value Range Propagation”. (*Computer Architecture*, Fall 02)
Implemented Value Range Propagation (using the concepts of control flow-graph, ssa-edges, ϕ -functions, variable renaming, loop carried dependence) for static branch prediction.
4. “Hardware Sorter”. (*Digital Hardware Design*, Spring 00)
Hardware implementation of bubble sort on *FPGA* to sort 8-bit numbers. (*Input*: sequential input to a RAM, *Output*: sequential display on 7-segment LED).
5. “Design and simulation of processor based on PowerPC architecture with extended instruction set” (*Computer Architecture*, Fall 99)
PowerPC based fully-pipelined (incorporated stalling and forwarding, and observed critical timing issues) computer architecture supporting an extended instruction set was designed and simulated.

Seminars and Poster Presentations:

- “DNAzymes and Aptamers”, *Duke Computer Science Nanotechnology Group Journal Club*, February 2007.
- “Synthetic Enzymes”, *Duke Computer Science Nanotechnology Group Journal Club*, November 2006
- “Protein Motor and Polymerases”, *Duke Computer Science Nanotechnology Group Journal Club*, October 2006.
- “Error Correction and Self-Healing in Self-Assembly”, *Duke Computer Science Nanotechnology Group Journal Club*, September 2006.
- “Design and Simulation of Self-Repairing DNA Lattices”, *inDuke Frontier2006*, *Duke University*, May 2006.
- “Optimal Algorithms for Oblivious Group Testing Problems”, *Algorithms Seminar, Department of Computer Science, Duke University*, March 2006.
- “Evolution of Random Graphs”, *Triseminar, Department of Computer Science, Duke University*, May 2005.
- “Self Assembly Model of DNA Tiles with Time-Dependent Glue Strength”, *Algorithms Seminar, Department of Computer Science, Duke University*, April 2005.
- “Self Assembly Model of DNA Tiles with Time-Dependent Glue Strength”, *Poster, DNA World Workshop, Caltech*, December 2004.
- “The Program-Size Complexity of Self-Assembly Squares”, *Duke Computer Science, Triseminar*, July 2004.
- “Compact Error Resilient Computational DNA Tiling Assemblies”, *Invited Talk, FNANO04, Snowbird, Utah*, April 2004.
- “Simulator for DNA nanorobotical devices”, *BICORN, Duke University Nanotech Group*, Summer 2003.

Graduate Courses:

Theory and Algorithms: *Design and Analysis of Algorithms, Complexity Theory, Randomized Algorithms, Approximation Algorithms, Numerical Analysis.*

Mathematics: *Applied Stochastic Processes, Topology, Combinatorics.*

Computational Bio and AI: *Algorithms in Computational Biology, Computational Functional Genomics, Numerical Artificial Intelligence.*

Nanotechnology: *Foundations of NanoSciences, Biomolecular Nanotechnology, Chem/bio Nucleic Acids, Physical Biochemistry.*

Systems and Architecture: *Advanced Computer Architecture.*

Undergraduate Courses:

Theory and Algorithms: *Data Structures, Analysis and Design of Algorithms, Theory of Computation, Discrete Structures, Logic for Computer Science, Programming Languages, Numerical and Scientific computing.*

Mathematics: *Analysis and Differential Equation, Complex Analysis and Vector Spaces, Optimization, Probability and Stochastic Processes.*

Graphics: *Computer Graphics, Digital Image Analysis.*

Systems and Architecture: *Computer Architecture, Digital Hardware Design, Operating Systems, File Structures and Information Systems, Computer Networks, DBMS.*

Professional Services

Organizing Committee Services

- 4th Conference on Foundations of Nanoscience: Self-assembled Architectures and Devices (FNANO07), Snowbird, Utah, 2007.
- 3rd Conference on Foundations of Nanoscience: Self-assembled Architectures and Devices (FNANO06), Snowbird, Utah, 2006.
- NSF workshop: Emerging Opportunities of Nanoscience to Energy Conversion and Storage, Nov 21-23, 2005, Washington DC.
- 2nd Conference on Foundations of Nanoscience: Self-assembled Architectures and Devices (FNANO05), Snowbird, Utah, 2005.
- 1st Conference on Foundations of Nanoscience: Self-assembled Architectures and Devices (FNANO04), Snowbird, Utah, 2004.

Reviewing Activities

- **Refree to Conferences:** Tenth International Meeting on DNA Computing (DNA10) 2004, Eighth International Conference on Knowledge Based Intelligent Information and Engineering Systems (KES) 2004, European Symposium on Algorithms (ESA) 2006.
- **Reviewer to Journals:** Theoretical Computer Science journal, J. of Chem. Technology and biotechnology, Physical Review Letters

Academic Achievements

- National Talent Search Scholarship from NCERT (National Council for Educational Research and Training) in 1996
- Rank 1 (among approximately 75,000 candidates) in MP-PET in 1998
- Rank 2 (among approximately 110,000 candidates) in DCE in 1998
- Rank 90 (among approximately 150,000 candidates) in IIT-JEE in 1998

- Certificate of merit from IIT Delhi in 1999
- Certificate of merit from CBSE in Class 10th (1996) and Class 12th (1998)

Computer Skills

Languages: C, C++, Perl, Java, Pascal, SML

Platforms: Unix, Linux, Windows, Solaris

Other skills: HTML, XML, Javascript, Matlab, CGI

Databases: PostgresQL, Oracle

Personal Information

- Date of Birth: August 27, 1980.
- Citizenship: Indian.
- Languages: English, Hindi.

References: Available on request.