IWBDA 2009

International Workshop on Bio-Design Automation

July 27, 2009
San Francisco, California, USA

46 YEARS
DESIGN AUTOMATION CONFERENCE

IWBDA 2009

1st International Workshop on
Bio-Design Automation

July 27, 2009

San Francisco, California, USA

Foreword

Welcome to the first International Workshop on Bio-Design Automation! IWBDA brings together researchers from the synthetic biology and design automation communities.

Still in its early stages, the field of synthetic biology has been driven by experimental expertise; much of its success has been attributable to the skill of the researchers in specific domains of biology. There has been a concerted effort to assemble repositories of standardized components. However, creating and integrating synthetic components remains an ad hoc process. The field has now reached a stage where it calls for computer-aided design tools. The electronic design automation (EDA) community has unique expertise to contribute to this endeavor. This workshop offers a forum for cross-disciplinary discussion, with the aim of seeding collaboration between the research communities.

The program consists of 13 talks and 16 poster presentations. These are organized into four sessions: "Models and Algorithms," "Experimental Biology," "Tools and Parts," and "Languages and Standardization." In addition, we are very pleased to have two of the most prominent researchers working at the intersection of engineering and biology as keynote speakers: Ron Weiss from MIT and Drew Endy from Stanford University.

We thank all the participants for contributing to IWBDA; we thank the Program Committee for reviewing abstracts; and we thank the Executive Committee, Soha Hassoun and Ion Mandoiu, for steering the workshop. Finally, we thank the Synthetic Biology Engineering Research Center, DNA 2.0, and Invitrogen for providing financial support.

Marc Riedel *Program Chair*
Douglas Densmore *General Chair*

Organizing Committee

General Chair	Douglas Densmore	UC Berkeley
Program Chair	Marc Riedel	Univ. Minnesota
Executive Committee	Soha Hassoun	Tufts Univ.
	Ion Mandoiu	Univ. Connecticut

Program Committee

J. Christopher Anderson, UC Berkeley
Adam Arkin, UC Berkeley
Douglas Densmore, UC Berkeley
Drew Endy, Stanford Univ.
Soha Hassoun, Tufts Univ.
Alfonso Jaramillo, Ecole Polytechnique
Yannis Kaznessis, Univ. of Minnesota
Eric Klavins, Univ. of Washington
Ion Mandoiu, Univ. Connecticut
Costas Maranas, Penn State Univ.
Chris Myers, Univ. of Utah
Marc Riedel, Univ. Minnesota
Herbert Sauro, Univ. of Washington
Christopher Voigt, UCSF
Ron Weiss, Princeton Univ.
Erik Winfree, Caltech
Chris Winstead, Utah State Univ.

IWBDA 2009 Program

Monday, July 27

8:00 - 9:00 : **Keynote Address (Room 124)**

Opening Remarks
Douglas Densmore, Marc Riedel, and Soha Hassoun

Keynote Address
Drew Endy

9:00 - 10:30 : **Session 1 - Models and Algorithms (Chair: Jaijeet Roychowdhury, Room 124)**

Synthesizing Sequential Register-Based Computation with Biochemistry
Adam Shea, Brian Fett, Marc Riedel, and Keshab Parhi

DOMINANT-EDGE PATHWAY: A Weighted Graph Algorithm for Identifying Dominant Metabolic Pathways
Ehasn Ullah, Kyongbum Lee, and Soha Hassoun

Simplified Biochemical Models using Factor Graphs
Chris Winstead and Chris Myers

Computer-Aided Synthetic Biology: How Multiscale Models can Rationalize the Design of Synthetic Gene Networks
Vassilis Sotiropoulos, Jonathan Tomshine, Katherine Volzing, Poonam Srivastava, and Yiannis Kaznessis

10:50 - 11:30 : **Session 2: Experimental Biology (Chair: Ron Weiss, Room 124)**

Automated Design of Synthetic Ribosome Binding Sites to Control Protein Production
Howard Salis, Ethan Mirsky, and Christopher Voigt

Programmed Control of Cellular Function: An in-cell Disease Prevention Device
Sangram Bagh, Mahuya Mandal, and David McMillen

11:30 - 12:00 : **Morning Poster Session (Rm 125)**

An Adaptive Data Structure for Biological System Design
Douglas Densmore, Bing Xia, Josh Kittleson, Timothy Ham, and J. Christopher Anderson

Applying Circuit Topological Analysis Techniques to Networks in Systems Biology
Sherief Reda

A Rigorous Approach to the Design of Oligonucleotides for PCR-based Gene Synthesis

P.J. Steiner

Bridging Synthetic Biology Models and Experiments using PoBoL
Michal Galdzicki, Deepak Chandran, Herbert Sauro, Daniel Cook, and John Gennari

Contamination Aware Droplet Routing for Digital Microfluidic Biochips
Tsung-Wei Huang and Tsung-Yi Ho

Designing Single-Duplex DNA Nanostructures by Abstraction
Shogo Hamada and Satoshi Murata

From Electronic to Biological Design Automation: Building the Bridge
Giuseppe Nicosia

12:00 - 1:30 : **Lunch**

1:30 - 2:15 : **Keynote Address (Room 124)**

Keynote Address
Ron Weiss

2:15 - 3:40 : **Session 3 - Tools and Parts (Chair: Herbert Sauro, Room 124)**

Design Tools for Synthetic Virology
Dimitris Papamichail and Steven Skiena

A Framework for Modeling Modular DNA Parts: Towards the Predictable Design of Synthetic Systems
Ty Thomson

Synthetic Biology: A New Application Area for Design Automation Research
Chris Myers, Nathan Barker, Kevin Jones, Hiroyuki Kuwahara, Curtis Madsen, Nam-Phuong Nguyen, and Chris Winstead

GenoCAD
Matthew William Lux, Yizhi Cai, and Jean Peccoud

Bridging Synthetic Biology Design and Experiments using PoBoL
Deepak Chandran, Michal Galdzicki, and Alec Nielsen

3:50 - 4:30 : **Session 4 - Languages and Standardization (Chair: Yiannis Kaznessis, Room 124)**

Towards a High-Level Programming Language for Standardizing and Automating Biology Protocols
Vaishnavi Ananthanarayanan and William Thies

BioBrick Open Language: A Keystone for Efficient Development and Communication of Standard Biological Parts
Ceasar Rodriguez, Doug Densmore, and Drew Endy

4:30 - 5:00 : **Afternoon Poster Session (Rm 125)**

Genetic Edge Detection
Jeff Tabor

JBEI Registry: Towards a Distributed Web of Registries
Timothy Ham, Zinovii Dmytriv, Paul Adams, and Jay Keasling

Overcoming Abstraction Barriers in Synthetic Biology Systems
Austin Che

PoBoL in the Lab: Structured Organization of Biological Samples for Laboratory Management and Automation
Alec Nielsen, Deepak Chandran, Michal Galdzicki, Sean Sleight, Herbert Sauro, Daniel Cook, and John Gennari

TinkerCell: a CAD application for Synthetic Biology
Deepak Chandran, Frank Bergmann, and Herbert M. Sauro

Toward Automatic Design of DNA Logic Gates and Devices
Ibuki Kawamata, Fumiaki Tanaka, and Masami Hagiya

Workflow Design for Synthetic Biology Using Clotho and Kepler
Douglas Densmore, Thien Nguyen, and J. Christopher Anderson

5:00 - 5:50 : **Panel Session (Moderator: Douglas Densmore, Rm 124)**

Panel Discussion
Lou Scheffer (HHMI and Cadence), Kevin Clancy (Invitrogen), Claes Gustafsson (DNA 2.0), and Ron Weiss (MIT)

Session 3 - Tools and Parts (Chair: Herbert Sauro, Room 124)

Session 4 - Languages and Standardization (Chair: Yiannis Kaznessis, Room 124)

Afternoon Poster Session (Rm 125)

Panel Session (Moderator: Douglas Densmore, Rm 124)

IWBDA 2009

Bios and Abstracts

Keynote Speaker: Drew Endy.

Drew Endy is currently an Assistant Professor in the Department of Bioengineering at Stanford University and a co-founder and President of the BioBricks Foundation (BBF). Drew Endy earned degrees in civil, environmental, and biochemical engineering at Lehigh and Dartmouth. He studied genetics & microbiology as a postdoc at UT Austin and UW Madison. From 1998 through 2001 he helped to start the Molecular Sciences Institute, an independent not-for-profit biological research lab in Berkeley, CA.

In 2002, he started a group as a fellow in the Department of Biology and the Biological Engineering Division at MIT; he joined the MIT faculty in 2004. Drew co-founded the MIT Synthetic Biology working group and the Registry of Standard Biological Parts, and organized the First International Conference on Synthetic Biology. In 2004 Endy co-founded Codon Devices, Inc., a biotechnology startup.

In 2005 Endy co-founded the BioBricks Foundation, a not-for-profit organization that is working to develop legal and economic strategies needed to support open biotechnology. Drew's research interests are the engineering of integrated biological systems and error detection & correction in reproducing machines.

Synthesizing Sequential Register-Based Computation with Biochemistry[1]

Adam Shea, Brian Fett, Marc D. Riedel, and Keshab Parhi

Department of Electrical and Computer Engineering
University of Minnesota
Minneapolis, MN 55455
Email: {shea0097, fett, mriedel, parhi}@umn.edu

Abstract

We presents a compilation strategy and a toolkit for biochemical reactions that perform sequential arithmetic computation on protein quantities, analogous to register-based computation in digital systems. From a Verilog-like input specification file, we generate biochemical reactions that produce output quantities as a function of input quantities, performing operations such as addition, subtraction, and multiplication. Sequential operations are implemented by transferring quantities between protein types, based on a clocking mechanism. Synthesis first is performed at a conceptual level, in terms of abstract biochemical reactions – a task analogous to *technology-independent logic synthesis* in circuit design. Then the results are mapped onto specific biochemical reactions, selected from libraries – a task analogous to *technology mapping* in circuit design. We demonstrate the algorithm on the synthesis of a variety of standard sequential functions: signal processing functions (FIR filters and IIR filters), vector multiplication, integration and differentiation. The designs are validated through transient stochastic simulation of the chemical kinetics. A possible experimental chassis is the universal DNA substrate developed by Erik Winfree's group at Caltech.

[1]This work is supported by the NSF CAREER Award #0845650 and by the Biomedical Informatics and Computational Biology program at the University of Minnesota.

DOMINANT-EDGE PATHWAY: **A Weighted Graph Algorithm for Identifying Dominant Metabolic Pathways**

Ehsan Ullah[▪], Kyongbum Lee[•], and Soha Hassoun[▪]
[▪]Department of Computer Science
[•]Department of Chemical and Biological Engineering
Tufts University, Medford, MA

Metabolic pathway analysis seeks to identify critical reactions in living organisms and plays an important role in synthetic biology. An algorithm, DOMINANT-EDGE PATHWAY, is presented that identifies a thermodynamically favored pathway from a specific substrate to metabolite end product in biochemical reaction network. The biochemical network is represented as a weighted directed graph based on the stoichiometry of the reactions. The search for the favored or dominant pathway is divided into two sub-problems. The first sub-problem, which resembles the bottleneck shortest path problem, involves finding the dominant path from a source metabolite, *s*, to one or more destination metabolites, *t*. We use a modified Dijkstra's algorithm to find the dominant-edge path. While visiting vertices, this algorithm models the effect of selecting favored reactions by including the maximum energy vertex into a frontier. Bottleneck series reactions are modeled by assigning a weight to candidate vertices equal to the minimum weight of the edge in the path from *s* to that vertex. The second sub-problem augments the partial dominant path to ensure it is stoichiometrically balanced. The algorithm identifies every dangling node *d* within the partial dominant path, and finds augmenting paths from every *d* to *t* using the modified Dijkstra's algorithm. These augmenting paths are merged with the partial dominant path to produce a stoichiometrically balanced dominant-edge pathway.

We have applied our algorithm to three test cases to demonstrate its efficiency. The algorithm's performance was compared to elementary flux mode (EFM) analysis (Schuster, 2000), a widely used metabolic pathway analysis tool. The first test case was the central carbon metabolic network of a microbe (*Zymomonas mobilis*) engineered to express four heterologous enzymes required for xylose utilization. (Altintas, 2006). The second test case was a recently published model of ethanol producing *Escherichia coli* (Trinh, 2008). The third test case was a model of the rat hepatocyte (liver cell) (Nolan, 2006).

Our algorithm departs from prior efforts to exhaustively enumerate all pathways in a reaction network. Given a desired pathway feature, e.g. thermodynamic favorability, our algorithm combines the weight assignment and the path identification steps. This combination results in a highly efficient search process compared to enumeration based approaches such as EFM and extreme pathway analysis. The algorithm is general with respect to the type of the desired pathway feature, and could be expanded to use other weights, e.g. measurement derived steady-state flux weights.

Simplified biochemical models using factor graphs

Chris Winstead[1] and **Chris J. Myers**[2]

[1]Dept. of Electrical and Computer Engineering, Utah State University
[2]Dept. of Electrical and Computer Engineering, University of Utah

Researchers are now developing synthetic genetic circuits to manipulate the biochemical processes within living cells. In order to model and predict the behavior of these circuits, the designer must account for numerous reactions among many chemical species and genetic components. There are now several computer analysis techniques and tools for predicting the function of genetic circuits, but at present they can only cope with small-scale reaction models [1]. *Model abstractions* are judicious approximations that reduce the complexity of biochemical models, thereby expanding the capabilities of computer-based analysis tools. This presentation introduces a possible framework for model abstraction based on factor graphs, which are now being used in numerous probabilistic signal processing applications [2].

To illustrate the possible applications of factor graphs to genetic circuits, example models are presented for a quorum-mediated trigger circuit [3], and for a circadian rhythm model [4]. In these examples, a factor graph is derived from the chemical master equation (CME) by factoring the system's joint probability distribution into approximately-independent factors. Each factor is treated as an independent sub-system in which appropriate local abstractions may be applied. This approach allows for hybrid simulations where the most effective models can be used for each reaction type in the system. The factor graph formalism maintains a clear chain-of-approximations that relates the simulation model to the CME, which is useful for evaluating model correctness. We propose that factor graphs will be useful for optimizing the complexity/accuracy tradeoff in more complex genetic circuit models.

References

[1] Daniel T. Gillespie. Stochastic simulation of chemical kinetics. *Annual Review of Physical Chemistry*, 58(1):35–55, 2007.

[2] H.-A. Loeliger, J. Dauwels, Junli Hu, S. Korl, Li Ping, and F.R. Kschischang. The factor graph approach to model-based signal processing. *Proceedings of the IEEE*, 95(6):1295–1322, June 2007.

[3] Chris Winstead, Nam Nguyen, and Chris J. Myers. A reliable quorum-mediated trigger circuit based on a genetic Muller C-element. In *Institute for Biological Engineering Symposium (IBE)*, Santa Clara, CA, March 2009.

[4] José M. G. Vilar, Hao Yuan Kueh, Naama Barkai, and Stanislas Leibler. Mechanisms of noise-resistance in genetic oscillators. *Proceedings of the National Academy of Sciences of the United States of America*, 99(9):5988–5992, 2002.

Computer-aided synthetic biology: How multiscale models can rationalize the design of synthetic gene networks.

Vassilis Sotiropoulos, Jonathan Tomshine, Katherine Volzing, Poonam Srivastava, Yiannis N. Kaznessis
Department of Chemical Engineering and Materials Science
University of Minnesota

Synthetic biology promises the development of novel gene networks that can enable precise control of protein expression. Although the successes of synthetic designs are spectacular, the paradigm of biological sciences as descriptive disciplines cannot assist in rationally engineering novel gene networks. A challenge facing the scientific and engineering communities is to reduce this enormous volume and complexity of biological data and dynamic behaviors into concise theoretical formulations with predictive ability that clearly associate synthetic DNA sequences to dynamic phenotypes.

A theoretical formulation for generating synthetic construct models would be a positive contribution if it were simple enough to be generally applicable; could be employed to build models for oscillators, bistable switches, logical gates, comparators, and the like; and could be codified in algorithmic form and made available to the community.

In this paper, we are describing such a theoretical formulation. We are developing algorithms that can generate and simulate models of arbitrary synthetic gene regulatory constructs [1-3]. We are combining experiments and simulations to determine the value and limits of this theoretical formulation and to validate it. We will discuss the following four topics:

i) Synthetic Biology Software Suite. We recently published version 1.0 of SynBioSS. *Designer* is a suite tool for automatically generating models of arbitrary synthetic constructs. Given a set of molecular components in a gene network (e.g. promoters, transcription factors, reporter genes, etc.), *Designer* applies universal molecular biology principles to build a set of biochemical reactions, including all reactions in transcription, translation, regulation and induction. SynBioSS *Wiki* is another suite component that allows users to store, retrieve and manipulate kinetic and equilibrium constants for biomolecular reactions.

ii) Development of multiscale dynamic models of gene expression systems. Representing synthetic gene networks with sets of reactions allows for a general method to be developed for constructing models. It also assists in connecting specific DNA sequences to targeted biological phenotypes. This truly rationalizes synthetic biology. However, these models are challenging to simulate numerically, because the systems are stiff (they are manifest over multiple scales) and away from the thermodynamic limit. We have produced multiscale algorithms that tackle stochastic behavior. We package all our algorithms in SynBioS*S Desktop Simulator (DS)*.

iii) Model-based design of synthetic regulatory networks. We model tetracycline-inducible networks and develop design principles for novel synthetic constructs. These systems are being used in important biomedical applications, and have been integral components of synthetic biology efforts.

iv) **Construct computer-aided designs of synthetic networks.** We experimentally construct tetracycline-inducible networks in *Escherichia coli*. In the presentation, we will illustrate how the combination of computer simulations and experiments can advance synthetic biology.

References

1) A. Hill, J. Tomshine, E. Wedding, V. Sotiropoulos, Y. Kaznessis, **"SynBioSS: the Synthetic Biology Modeling Suite"**, Bioinformatics 2008, 24(21):2551-2553
2) Y.N. Kaznessis **"Models for synthetic biology."** BMC Syst Biol. 2007 Nov 6;1(1):47
3) V. Sotiropoulos, Y. Kaznessis, **"Computer-aided design of synthetic tetracycline-inducible regulatory networks"**, BMC Systems Biology, 2007, 1:7

Automated Design of Synthetic Ribosome Binding Sites to Control Protein Expression

Howard M. Salis[1], Ethan A. Mirsky[2], and Christopher A. Voigt[1]

[1]Department of Pharmaceutical Chemistry, University of California San Francisco, San Francisco, CA, 94158

[2]Graduate Group in Biophysics, University of California San Francisco, San Francisco, CA, 94158

Email: salish@picasso.ucsf.edu / ethan.mirsky@ucsf.edu / cavoigt@picasso.ucsf.edu

Abstract

Engineering genetic systems requires a fine control over a protein's expression level; for example, to connect genetic circuits or control flux through a metabolic pathway. We have developed a predictive design method that generates a synthetic ribosome binding site sequence to achieve a *user-selected* protein expression level. The design method combines a predictive thermodynamic model of bacterial translation initiation with a Monte Carlo optimization algorithm. Experimental validation of the design method in *Escherichia coli* shows that the method is accurate to within a factor of about two over a range of control from 1 to 100 000 on a proportional scale. The design method also correctly predicts that reusing a ribosome binding site with different proteins can result in dramatically different expression levels, explaining why these genetic parts do not function in a modular fashion. The proposed forward engineering approach can be combined with a systems-level model of a genetic circuit or metabolic network to rationally design and optimize their function.

Programmed control of cellular function: An in-cell disease prevention device

Sangram Bagh[1], Mahuya Mandal[2] and David R. McMillen[3]

Institute of Optical Science and Department of Chemical and Physical Sciences
University of Toronto at Mississauga, Mississauga, ON L5L 1C6 Canada

E-mail: [1]sangram.bagh@utoronto.ca , [2]mahuya.mandal@gmail.com ,
[3]david.mcmillen@utoronto.ca

Abstract: One of the distant dreams of synthetic biologist is to develop a programmed gene circuit, which could act as automatic in-cell disease detection and prevention device that would stay dormant in normal condition but could become active to prevent fatal disease upon detection (e.g. in cell cancer detector). In this work, adopting simple engineering principle, we developed synthetic genetic controllers, which work as a programmed intra-cellular device to detect and prevent fatal viral effects in bacteria. Bacteriophage lambda is a virus, which attacks *Escherichia coli* and establishes a bi-stable lysis-lysogeny switch. Lysogeny is no-harm state and lysis is the state where the bacterium dies. When lysigenised K-12 strain of *Escherichia coli* is irradiated with UV, the cell irreversibly enters in lytic pathway. We have designed, constructed and implemented synthetic genetic controllers in lysogenised K-12 strain of *Escherichia coli* to prevent the lysis in a programmed way. Our experimental results showed that those controllers stayed dormant in lysogeny state, detected the change in biochemical signal in UV induced cell, turned on automatically to prevent lysis, and turned off on demand by changing temperature of the system. A mathematical model has also been constructed to capture the essential features of the controllers coupled with K-12 cell. The combined experimental and theoretical analysis showed that the performance of a controller depends on the nature of the promoter, different kinetic parameters of the system and copy number of the controller. Our experimental and computational results demonstrate that those controllers work as in cell disease detection and prevention device in K-12 *Escherichia coli*.

An Adaptive Data Structure for Biological System Design

Douglas Densmore[1], Bing Xia[1], Josh Kittleson[2], Timothy Ham[3], J. Christopher Anderson[2]
Department of Electrical Engineering and Computer Sciences[1]
Department of Bioengineering[2]
University of California, Berkeley
Joint BioEnergy Institute[3]

densmore@eecs.berkeley.edu, aix.bing@gmail.com,
josh.kittleson@gmail.com, tsham@lbl.gov, jcanderson@berkeley.edu

Establishing standards are a key requirement for new a field to gather momentum. In addition they provide a path for tool developers to create offerings with maximum interoperability. One needed standard for synthetic biology is the development of a core data model. This model should encapsulate information about biological designs which not only provides a way for the data to be represented but also establishes a relationship between data elements which tools can exploit.

While the development of a core data model is the ultimate goal, tools must be created concurrently while this data model is under development. Therefore we need a data structure which provides a meaningful API to tools while at the same time remaining flexible to changes in the data model specification.

This work presents the development of a core data model by researchers at UC Berkeley and the Joint BioEnergy Institute (JBEI). The goal is to create a data model which is generic enough to represent a large design space but specific enough to be useful to those actually carrying out experimental work. Additionally it should be extensible and not be specific to any one current assembly standard.

To illustrate these concepts, we present both the data model organization visually (as a collection of related objects and fields) as well as the data structure implementation in the Clotho [1] design framework. Clotho's implementation is specifically interesting as it is generated dynamically at runtime and used to provide a robust API which is constant in the face of a changing data model specification.

In addition to the data model, data structure, and API, we also provide a number of design scenarios which are proposed as a benchmark by which the data model and subsequent modifications to it will be evaluated. These will be vital to the community as whole.

[1] http://biocad-server.eecs.berkeley.edu/wiki/index.php/Tools

Applying Circuit Topological Analysis Techniques to Networks in Systems Biology

Sherief Reda, Division of Engineering, Brown University, Providence, RI 02912.
E-mail: sherief_reda@brown.edu

The main purpose of digital circuits is information processing which is carried out using logic gates and intercommunication wires. The cell is an integrated device made of thousands/millions of interacting proteins. The cell monitors its internal and external environment and accordingly responds using different proteins that exercise various cell functionalities. The interactions among genes, proteins and various signal molecules form complex information processing networks that map the cell's signals and molecules into functions. Cellular networks are at the center of the study of the field of systems biology [1].

Networks arising from electronic circuits and systems biology contexts share many similarities. At their core they both process information. Much of this information processing is carried out in systems biology in a logical manner (e.g., using conjunction, disjunction and negation operators) similar to digital circuits. In both networks computation occurs distributively at the nodes of the network and communication is carried out between the nodes of the network to realize the network's function. In electronic circuits, metal wires transfer electrons in the circuit for communication purposes. In systems biology, proteins and various biochemical molecules floating in the water-based medium of the cell carry out the required communication. Just as nodes in circuit networks can be grouped into modules (e.g., flip-flops, adders) based on their functionalities, nodes in systems biology can be grouped into *motifs* that are used to carry out specific recurring functionalities (e.g., feed-forward loops and autoregulation).

Given the similarities between systems biology networks and electronic circuits, this work seeks to investigate the applicability of circuit topology and structure analysis techniques to systems biology networks. In particular, electronic circuits have shown unique properties as displayed by the famed Rent's rule [2]. Rent's rule is a power-law relationship that holds for computing circuitry. The rule relates the number of external wires emanating from a block of computational cells to the number of cells within the block. The rule has been observed and validated on many real circuit designs. It has many applications in circuit design and implementation including wirelength estimation, congestion estimation and interconnect power estimation.

This work seeks to investigate the following questions: Does Rent's rule also extend to systems biology networks? What are the structural similarities and differences between systems biology networks and electronic circuits? Did evolution lead to systems biology network that share similarities (e.g., hierarchy) with designed electronic circuits? In this work we investigate the applicability of circuit topological analysis techniques for networks arising in systems biology. In particular, we investigate the following

- We study the applicability of Rent's rule for systems biology networks. We explain the topological similarities and differences between electronic circuits and cellular networks in systems biology.
- To provide a basis for comparing different networks, we utilize some of the latest results in graph theory to construct random networks with the same number of edges, nodes and node degree sequences as the experimented real networks. We investigate the applicability of Rent's rule to these random networks.
- We provide comprehensive experimental results on large selection of representative networks from systems biology including protein-protein networks and transcription networks. We classify these networks based on their functionality, and we show that the Rent's exponent range depends on the functionality of the network. Networks that process information in a logical fashion have Rent exponents that are in the same range as in electronic circuits.

References

[1] U. Alon, *An Introduction to Systems Biology: Design Priniciples of Biological Circuits*. Chapman & Hall/CRC, 2007.

[2] B. Landman and R. Russo, "On a Pin versus Block Relationship for Partitions of Logical Graphs," *IEEE Transactions on Computers*, vol. 20(C), pp. 793–813, 1971.

A Rigorous Approach to the Design of Oligonucleotides for PCR-based Gene Synthesis

PJ Steiner
MIT CSAIL, Knight Lab
pjs@csail.mit.edu

Abstract

The most fundamental task in synthetic biology is gene synthesis, the *de novo* synthesis of novel DNA. Gene synthesis can already be outsourced to a number of companies, and there already exists software to aid the synthetic biologist who wishes to perform syntheses himself [1, 2, 3, 4]. However, gene synthesis services are opaque, and the available software is neither sufficiently flexible nor computationally rigorous.

Most often, DNA is synthesized from overlapping oligos by ligation or PCR. The primary computational problem in gene synthesis is the selection of the optimal set of oligos to assemble a given sequence. This problem is made difficult by its large solution space and by the presence of a number of constraints: one must work around repetitive structures which can cause misassembly while keeping the melting temperatures of the oligo overlaps uniform.

This work presents a rigorous approach to the design of PCR-based gene syntheses. It describes an algorithm which finds an optimal set of oligos for the synthesis of a given sequence: the set of oligos with maximally uniform overlap melting temperatures that satisfies constraints which prevent mispriming. The algorithm is graph-based: the optimal solution corresponds to the shortest path through a graph constructed from the constraints. A software tool employing the algorithm has been implemented in Common Lisp.

The space of potential sets of oligos is even larger when the DNA to be synthesized contains protein-coding regions — because the genetic code is degenerate, a combinatorial number of different sequences code for the same protein. If the primary concern is the protein the sequence codes for, codons can be changed to synonymous codons without consequence, making it possible to remove problematic repetitive elements. We show how the combination of the above algorithm and techniques from constraint optimization could allow us to systematically change codons in order to improve the properties of a sequence to be synthesized.

[1] David M. Hoover and Jacek Lubkowski. DNAWorks: an automated method for designing oligonucleotides for PCR-based gene synthesis. *Nucleic Acids Research*, 30(10):e43, 2002.

[2] Sebastian Jayaraj, Ralph Reid, and Daniel V. Santi. GeMS: an advanced software package for designing synthetic genes. *Nucleic Acids Research*, 33(9):3011–3016, 2005.

[3] Jean-Marie Rouillard, Woonghee Lee, Gilles Truan, Xiaolian Gao, Xiaochuan Zhou, and Erdogan Gulari. Gene2Oligo: oligonucleotide design for in vitro gene synthesis. *Nucleic Acids Research*, 32(Web Server Issue):W176–W180, 2004.

[4] Lance Stewart and Alex B. Burgin. Whole Gene Synthesis: A Gene-O-Matic Future. *Frontiers in Drug Design & Discovery*, 1(1):297–341, 2005.

Title: Bridging Synthetic Biology Models and Experiments using PoBoL

Authors:
Michal Galdzicki[1] <mgaldzic@u.washington.edu>
Deepak Chandran[2] <deepakc@u.washington.edu>
Herbert M. Sauro[2] <hsauro@u.washington.edu>
Daniel L. Cook[1,3] <dcook@u.washington.edu>
John H. Gennari[1] <gennari@u.washington.edu>

Affiliations:
[1]Biomedical & Health Informatics, University of Washington, Seattle, WA 98195, USA
[2] Bioengineering, University of Washington, Seattle, WA, 98195, USA
[3] Physiology & Biophysics, University of Washington, Seattle, WA, 98195, USA

Abstract:
To facilitate exchange of information between computer applications used in synthetic biology we propose the use of a semantic model representation. The Provisional BioBricks Language (PoBoL) is a proposed electronic data exchange standard with the capability to serve this role (BioBricks Foundation Request For Comments 31). As synthetic biology employs diverse methodologies from computational modeling, information systems, and experimental techniques, there is a need for a common language to encode, store, and access data created by disjoint software tools. Simulations of synthetic circuits using computational models are necessary to understand the dynamics and optimize their design. Data repositories that house information about available genetic parts and their properties assist the reuse of effective designs. Finally, optimized experimental procedures help the reliable construction of new biological systems in the laboratory. Representation of such elements using PoBoL will facilitate the efficient exchange of information between these three pillars of synthetic biology and will pave the way for efficient design and engineering of biological circuits. Computational access across models, part repositories, and the experimental protocols can provide the information needed to synthesize or assemble the physical system directly. In order to achieve such access, biological parts must be represented in a format structured to support computational processing. This format should be sufficiently flexible to contain information for the experimentalist as well as the modeler. Such information might include the DNA sequence of the part, assembly information, parameters describing the part's dynamics, and annotation information. In order to take advantage of generic software tools and common network infrastructure The Web Ontology Language (OWL) provides a W3C supported technology capable of expressing the complex description of the underlying data model. In future work we plan to extend PoBoL to verify quality of data and promote the use of modular design. Use, adoption, and continued development of PoBoL to accomplish these ambitions should help the broader synthetic biology community.

Contamination Aware Droplet Routing for Digital Microfluidic Biochips

Tsung-Wei Huang and Tsung-Yi Ho, *Member*, *IEEE* *

Abstract

Recent advanced in digital microfluidic biochips (DMFBs) have revolutionized the traditional laboratory procedures. By providing the droplet-based system, DMFB can achieve real-time biological analysis, and safety-critical biomedical applications. However, the contamination problem caused by bead retention and liquid residue has become a key design problem for droplet routing stage. This is because the liquid residues may cause an inevitable erroneous reaction between different bio-molecules, which leads to a complete malfunction for bioassay. To overcome this problem, a wash droplet is introduced to clean the contaminations on the surface of the microfluidic array. In this paper, we propose a contamination-aware droplet routing algorithm for DMFBs. Compared with the state-of-the-art technique, experimental results demonstrate that our contamination aware droplet routing algorithm achieves significant better results for minimizing the contaminated spots, the number of used cells, and the latest arrival time.

1 Background and Related Prior Work

The basic cell of a digital microfluidic biochip (DMFB) consists of two parallel glass plates. The bottom plate contains a patterned array of individually controllable electrodes, and the top plate is coated with a continuous ground electrode. The droplets containing biochemical samples, and the filler medium, such as silicone oil, are sandwiched between the plates. By varying the electrical potential along a linear array of electrodes, droplets can be moved along this line of electrodes due to the principle of electrowetting on dielectric (EWOD) [1, 6]. The basic operations (e.g., dilute, mix, etc.) can be performed anywhere in the 2D microfluidic array because each basic cell has the same architecture. Besides the 2D microfluidic array, there are on-chip reservoirs/dispensing ports and optical detectors. The dispensing port/reservoirs are responsible for droplet generation while the optical detectors are used for droplet detection. Electrodes are connected to control pins for electrical actuation. Therefore, by controlling voltage to each electrode in the bottom glass plate with control pins, we can control the moving direction of droplets.

Droplet routing is a critical step in DMFB physical design automation. Unlike traditional VLSI routing, in addition to routing path selection, the droplet routing problem needs to address the issue of scheduling droplets under the practical constraints imposed by the fluidic property and the timing restriction of the synthesis result. Droplet routing problem has attracted much attention in the literature recently [4, 7, 8]. Current droplet-routing algorithms have a detrimental effect on the unrestricted sharing of used cells by various droplet routes. Therefore, contaminations between droplets is inevitable either within one sub-problem or between sub-problems.

Contaminations caused by bead retention and liquid residue between droplet routes of different biomolecules may cause inevitable erroneous reaction. Moreover, these errors will possibly breakdown the electrodes and cause electrode short problems, which results in physical defects and produces incorrect behaviors in the electrical domain. Although silicone oil with its low surface tension and spreading property has been advocated as a filler medium to prevent contaminations, it has been proved that it is not sufficient for many types of proteins and heterogeneous immunoassays[5]. To cope with this problem, a wash droplet is introduced to clean the contaminated spots on the surface of the microfluidic array.

In [9], a novel droplet routing algorithm is proposed for cross-contamination avoidance. It attempts to determine disjoint droplet routes by modified Lee algorithm. If a disjoint droplet route does not satisfy the timing constraint, a greedy wash-droplet routing is performed based on an optimization model. However, droplets routed in disjoint order may increase the number of used cells significantly that degrade reliability and fault tolerance for bioassays. Furthermore, the greedy wash-droplet routing within one sub-problem and between sub-problems may increase the latest arrival time significantly.

Thus, if wash droplets are not considered with droplets, the latest arrival time of a sub-problem will increase significantly that degrade reliability and slow execution for bioassays. Furthermore, not only intra-contaminations occur within one sub-problem but also inter-contaminations between sub-problems. Additional wash droplets are needed to clean the contaminations between sub-problems and add additional execution time for bioassays.

2 Current Results

In this paper, we present a contamination-aware droplet routing algorithm for DMFBs. We first constructs the global routing tracks by analyzing the preferred moving direction of each droplet to guide the A* maze searching. After that, we adopt a K-shortest path routing technique to minimize the contaminated spots [2]. To fully consider contaminated spots within one sub-problem, a routing compaction technique by dynamic programming is used to determine the contaminated spots within one sub-problem. As for contaminated spots between sub-problems, a look-ahead contamination prediction technique is used to determine the contaminated spots between sub-problems. After that, a minimum cost circulation technique is adopted to simultaneously clean intra- and inter-contaminations to minimize the used cells and the latest arrival time [3]. Unlike the greedy wash-droplet routing proposed in [9], our algorithm has the following distinguished features:

- A global moving vector analysis for constructing preferred routing tracks to minimize the number of used unit cells.
- A K-shortest path routing technique to minimize the contaminated spots within one sub-problem.
- A routing compaction technique by dynamic programming to determine the contaminated spots within one sub-problem and minimize the latest arrival time of droplets.
- A look-ahead contamination prediction technique to determine the contaminated spots between sub-problems and minimize the total routing time of wash droplets.
- A minimum cost circulation technique is adopted to simultaneously clean intra- and inter-contaminations to minimize the used cells and the latest arrival time.

Compared with the state-of-the-art technique [9], experimental results demonstrate that our contamination aware droplet routing algorithm achieves significant better results for minimizing the contaminated spots, the number of used cells, and the latest arrival time.

References

[1] S. K. Cho, S.-K. Fan, H. Moon, and C.-J. Kim, "Towards digital microfluidic circuits: Creating, transporting, cutting and merging liquid droplets by electrowetting-based actuation," *Proc. MEMS Conf.*, pp. 32–35, Jan. 2002.

[2] D. Eppstein, "Finding the k shortest paths," *Proc. IEEE FOCS*, pp. 154–165, Feb. 1994.

[3] A. V. Goldberg and R. E. Tarjan, "Finding minimum cost circulations by canceling negative cycles," *J. ACM* , pp. 873–886, Oct. 1989.

[4] E. J. Griffith, S. Akella, and M. K. Goldberg, "Performance characterization of a reconfigurable planar-array digital microfluidic system," *IEEE Trans. on CAD*, vol. 25, no. 2, pp. 345–357, Feb. 2006.

[5] H. Moon, A. R. Wheeler, R. L. Garrell, J.A. Loo, and C. J. Kim, "An integrated digital microfluidic chip for multiplexed proteomic sample preparation and analysis by MALDI-MS," *Lab on a Chip*, vol. 6, pp. 1213–1219, Feb. 2006.

[6] M. G. Pollack, A. D. Shenderov, and R. B. Fair, "Electrowetting-based actuation of droplets for integrated microfluidics," *Lab Chip*, vol. 2, no. 2, pp. 96–101, May 2002.

[7] F. Su, W. Hwang, and K. Chakrabarty, "Droplet routing in the synthesis of digital microfluidic biochips," *Proc. IEEE/ACM DATE*, pp. 1–6, Mar. 2006.

[8] P. H. Yuh, C. L. Yang, and Y. W. Chang, "BioRoute: A network flow based routing algorithm for the synthesis of digital microfluidic biochips," *IEEE Trans. on CAD*, vol. 27, no. 11, pp. 1928–1941, Nov. 2008.

[9] Y. Zhao and K. Chakrabarty, "Cross-contamination avoidance for droplet routing in digital microfluidic biochips," *IEEE/ACM DATE*, Apr. 2009.

*Contact Author: Tsung-Yi Ho is with the Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan 701, Taiwan; tyho@csie.ncku.edu.tw

Designing Single-Duplex DNA Nanostructures by Abstraction

Shogo Hamada[1,2] and Satoshi Murata[1]

1. Dept. of Computational Intelligence and Systems Science, Tokyo Institute of Technology, Yokohama, 226-8502 Japan

2. JSPS Research fellow

E-mail: hamada@mrt.dis.titech.ac.jp, murata@dis.titech.ac.jp

In addition to the role of genetic information career in nature, DNA molecule is also well known as a nanoscale material to create 2-D and 3-D nanostructures. Although diverse studies achieved a wide variety of DNA nanostructures for various shapes and patterns[1-4] so far, still our design possibilities are somewhat limited. The first reason for this is the lack of structural varieties of DNA junctions. Currently, almost all structures are based on a limited choice of DNA crossover junctions. This confinement in junction variety gives us a limitation in the structural geometry. The second reason is the lacking of standardized abstraction for design. Although such abstractions are introduced in some studies like 2-D/3-D DNA Origami[5,6], still most of structures are designed only by ad-hoc approach. Here we demonstrate an interconnected single-duplex DNA junction and designs of various structures based on the junction by graphical abstraction. Our new "right-angled" junction enables the organization of DNA helices into broader geometry, which gives us wider choice of DNA nanostructures. In addition, our abstraction technique for motif design is based on a half helix of DNA as one structural unit, so it can be easily applied to other junctions and can combine them as well. This block-based abstraction will make us simplify the design process of DNA motifs, which leads to the standardized method of designing DNA nanostructures.

References

[1] Fu, T-J. & Seeman, N. C. DNA double-crossover molecules. *Biochemistry.* **32**(13), 3211–3220 (1993).

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

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

[4] He, Y., Chen, Y., Liu, H., Ribbe, A. E. & Mao, C. Self-assembly of hexagonal DNA two-dimensional (2D) arrays. *J. Am. Chem. Soc.* **127**(35) 12202–12203 (2005).

[5] Rothemund, P. W. K. Folding DNA to create nanoscale shapes and patterns. *Nature.* **440**(7082) 297–302 (2006).

[6] Douglas, S. M., Dietz, H., Liedl, T., Högberg, B., Graf, F. & Shih, W. M. Self-assembly of DNA into nanoscale three-dimensional shapes. Nature. **459**(7245) 414-418 (2009).

From Electronic to Biological Design Automation: Building the Bridge

Giuseppe Nicosia
Department of Mathematics and Computer Science
University of Catania – Viale A. Doria, 6, 95125, Catania, Italy
nicosia@dmi.unict.it
http://www.dmi.unict.it/nicosia

This research line seeks to create a software platform for modelling, optimization, simulation and programming of cellular and molecular systems by integrating proposed new computational tools with existing state-of-the-art methodologies, adopted from the engineering domain of Electronic Design Automation (EDA). This research stresses the inter- intra-disciplinary nature of molecular biology, synthetic biology, systems medicine and personalized healthcare. The research is aimed at explaining biology through a common language framework analogous to EDA and by real examples operating at different levels. These levels in biological terms are genes, RNAs, peptides, proteins, large molecular entities, biological pathways, molecular and cellular interactions, tissue formation, organ level, organism, and patient.

The time is mature to exploit the following natural analogies between the two areas: 1) *molecular entities* correspond to *electronics devices*; 2) *biological pathways* correspond to *electronics circuits*; 3) *molecular and cellular interactions* correspond to *coupling among analog and digital circuits*; 4) *tissues* can be viewed as *electronics systems* (e.g., board); 5) *an organ* could correspond to *highly complex control and communication system*. My vision is to adopt the existing methodologies (e.g., mathematical modelling, optimization algorithms, circuit sizing, design-for-yield, model order reduction techniques, stochastic simulations, sensitivity analysis methods, robust design and test, high-level synthesis) originally developed for the design and fabrication of electronics parts, so that they can be used effectively to *design and manufacture advanced and useful biological parts*. Thus the goal of this research line is to build a new paradigm and framework for the bio-fab and the new healthcare.

In this talk I will show examples focusing on each of these design layers, and assess the strengths and weaknesses of tools for modelling, optimizing, simulating and programming at different levels. These design layers, in terms of EDA will benchmark information processing and computational features of the tools as they apply to each biological level. As a consequence, it is mandatory to face the challenge of creating suitable abstraction tools, approached through analogies and by examples, in order to capture various information processing capabilities. It is necessary to study new tools for the biologist and physician to model and simulate various aspects of molecular biology. Using several suite of tools: namely, UML, Simpathica, BIOSPICE, SPIM, BetaSIM, etc. I am constructing and analyzing models of biological systems. These models are at various levels: the biochemical level using SPiM or BlenX, the molecular and cellular level using BetaSIM and Simpathica, the molecular networks using little B, the tissue level using BIOSPICE, and finally the organ level using UML. This strategy allow me to employ a bottom-up approach as commonly used in a complex systems analysis and design, and avoids all the shortcomings of the top-down approach used in a reductionist fashion, while combining both approaches in a single agnostic methodology. Moreover, I will show reverse engineering techniques, machine learning and optimization algorithms in order to program specific molecular systems with a particular target behaviour (e.g., steady state behaviour).

Finally, for a given biological model in a given layer, the goals of the proposed framework are:

a) Sensitivity Analysis: to detect *the sensitive parameters* of the given model and *the most important biological entities and parts*;

b) Circuit Sizing: maximize the number of *satisfied biological properties* (and, as long term goal, the *physiological property*);

c) Robustness Analysis: maximize the *robustness of the target behaviour and properties*;

d) Design-for-Yield: maximize the *yield of the designed biological parts*;

e) Multi-Criteria Decision Making: to select the "best", *Pareto optimal, biological circuit(s) in terms of number of satisfied properties, behaviour, robustness and yield*.

Keynote Speaker: Ron Weiss.

Ron Weiss is an Associate Professor in the Department of Biological Engineering and in the Department of Electrical Engineering and Computer Science at the Massachusetts Institute of Technology. He received his PhD from MIT in 2001 and held a faculty appointment at Princeton University between 2001 and 2009. His research focuses primarily on synthetic biology, where he programs cell behavior by constructing and modeling biochemical and cellular computing systems. A major thrust of his work is the synthesis of gene networks that are engineered to perform *in vivo* analog and digital logic computation. He is also interested in programming cell aggregates to perform coordinated tasks using cell-cell communication with chemical diffusion mechanisms such as quorum sensing. He has constructed and tested several novel *in vivo* biochemical logic circuits and intercellular communication systems. Weiss is interested in both hands-on experimental work and in implementing software infrastructures for simulation and design work. For his work in synthetic biology, Weiss has received MIT's Technology Review Magazine's TR100 Award ("top 100 young innovators", 2003), was selected as a speaker for the National Academy of Engineering's Frontiers of Engineering Symposium (2003), received the E. Lawrence Keyes, Jr./Emerson Electric Company Faculty Advancement Award at Princeton University (2003), his research in Synthetic Biology was named by MIT's Technology Review Magazine as one of "10 emerging technologies that will change your world" (2004), was chosen as a finalist for the World Technology Network's Biotechnology Award (2004), and was selected as a speaker for the National Academy of Sciences Frontiers of Science Symposium (2005). Over the last few years, Weiss has had several major publications in journals such as Nature, Nature Biotechnology, and PNAS.

Design Tools for Synthetic Virology

Dimitris Papamichail[1], Steven Skiena[2]

[1] Computer Science Department, University of Miami, Coral Gables, FL 33146, USA.
Email: dimitris@cs.miami.edu
[2] Computer Science Department, Stony Brook University, Stony Brook, NY 11794, USA.
Email: skiena@cs.sunysb.edu

In recent years synthetic genomics has shown a great potential for advancing science; from reading to writing the genome code and synthesizing organisms and cells that make useful products, such as pharmaceuticals, vaccines, substitutes for petrochemicals and bio-energy, as well as using renewable sources such as sunlight, CO_2 and plant biomass.

The first genome-level synthesis of a virus (poliovirus) by our collaborators Cello, Paul, and Wimmer [1] received the attention of the world's press on its publication in Science in July 2002. Since then, other viruses and even small bacterial genomes have been synthesized. Biotechnology has reached the point where artificial chromosomes of limited length (approximately 500K base pairs) can be created with high fidelity and cost that is exponentially decreasing.

Our group has designed, synthesized, and evaluated four new variants of poliovirus to serve as vaccines [2,3]. Specifically, we sought weakened but viable strains that may be used for preparations of a killed poliovirus vaccine, a point of tremendous significance in the context of the ongoing global polio eradication. Our designs result in a virus with roughly 100-fold lower specific infectivity than the wildtype virus. The most exciting thing about our technique is that it suggests an approach to construct vaccines for essentially any viral disease, including those for which vaccines have not yet been developed.

To achieve these designs, we have developed tools and algorithms for embedding/removing patterns, constraining secondary structures, overlapping coding frames, and adhering to pre-specified codon and codon-pair distributions. These complement a variety of existing tools that optimize designs for manufacturability (i.e. oligos without local secondary structures and end repeats), while we are optimizing sequences for biological activity. Challenges arise in realizing heterogeneous combinations of sequence design preferences and constraints. Such combinations lead to NP-complete problems, for which we employ simulated annealing techniques.

Currently we are working on a second generation of design tools, which will feature a friendly user interface and interconnect with publicly available databases, providing access to a variety of genes, transcription factors and pathways. This ever growing wealth of genomic information will serve as design components for the creation of cells and microorganisms for specific uses, based on the kind of metabolism one wants or the functions that should be performed.

References

[1] J. Cello, A. Paul, and E. Wimmer, “Chemical Synthesis of Poliovirus cDNA: Generation of Infectious Virus in the Absence of Natural Template”, *Science* 297 (2002) 1016-1018
[2] J. R. Coleman, D. Papamichail, S. Skiena, B. Futcher, E. Wimmer and S. Mueller, Virus attenuation by genome-scale changes in codon-pair bias: a novel method for developing viral vaccines, *Science,* Vol. 320, no. 5884, pp. 1784 - 1787, 27 June 2008.
[3] S. Mueller, D. Papamichail, J.R. Coleman, S. Skiena and E. Wimmer, Reduction of the rate of poliovirus protein synthesis through large scale codon deoptimization causes virus attenuation of viral virulence, *Journal of Virology*, Vol. 80, No. 19, pp. 9687-9696, October 2006.

A framework for modeling modular DNA parts: Towards the predictable design of synthetic systems

Ty Thomson
ty@plectix.com
Plectix BioSystems, Inc.,
20 Holland St, Suite 400, Somerville, MA 02144, USA

The BioBrick standard was designed to facilitate the easy construction of a diverse array of synthetic systems by optimizing the modularity and reusability of functional DNA parts. This is made possible by a strong adherence to the principle of encapsulation in the design of the individual BioBrick parts, wherein each component is intended to be functionally independent of the other parts in the system in which it is used. However, there is currently no standard *in silico* framework for modeling systems composed of modular DNA parts that also offers this same level of functional reusability. We have developed a conceptual framework for modeling BioBrick parts that closely mirrors the modularity of the parts themselves. Importantly, this framework can be implemented in an array of widely available rule-based modeling tools, allowing the synthetic biology field to benefit from software and methods developed by the broader systems biology community. Additionally, the modular and reusable architecture makes it easy, even for non-modelers, to construct a model of a complete system if there are pre-existing models for the constituent parts. A standardized framework for modeling synthetic systems would allow for the rapid computational prototyping of synthetic systems enabling researchers to design, explore, test and optimize the behavior of synthetic systems prior to their actual construction in the laboratory.

Synthetic Biology: A New Application Area for Design Automation Research

Chris J. Myers[1], Nathan Barker[2], Kevin Jones[1], Hiroyuki Kuwahara[3], Curtis Madsen[4], Nam-Phuong D. Nguyen[5], Chris Winstead[6]

[1]Dept. of Electrical and Computer Engineering, University of Utah,
[2]Dept. of Computer Science and Information Systems, Southern Utah University,
[3]Microsoft Research - U. of Trento Centre for Computational and Systems Biology,
[4]School of Computing, University of Utah,
[5]Dept. of Computer Science, University of Texas in Austin,
[6]Dept. of Electrical and Computer Engineering, Utah State University,

Electronic design automation (EDA) tools have facilitated the design of ever more complex integrated circuits each year. *Synthetic biology* would also benefit from the development of *genetic design automation* (GDA) tools. Synthetic biology has the potential to help us produce drugs more economically [7], metabolize toxic chemicals [3], and even modify bacteria to hunt and kill tumors [1]. There are, however, numerous challenges when designing the genetic circuits used in these applications. First, existing GDA tools require biologists to design and analyze genetic circuits at the molecular level, roughly equivalent to the layout level for electronic circuits. Another serious challenge is that genetic circuits are composed of very noisy components making their behavior more asynchronous, analog, and non-deterministic in nature. New GDA research is necessary to address these challenges. Interestingly, future electronic circuits may soon also face many of the same challenges. This fact opens up the very intriguing idea that this research may in the future also be utilized to produce more robust and power efficient electronic circuits. This talk gives an overview of some of the key concepts necessary for EDA engineers to contribute to this area of research which are highlighted in a new textbook, *Engineering Genetic Circuits* [5]. This talk also describes our development of `iBioSim`, a GDA tool that supports higher levels of abstraction for modeling, analysis, and design of genetic circuits [2, 4, 6]. Finally, this talk presents some of the important theoretical and computational research problems in this area that must be addressed.

References

[1] J. C. Anderson, E. J. Clarke, and A. P. Arkin. Environmentally controlled invasion of cancer cells by engineering bacteria. *Journal of Molecular Biology*, 355:619–627, 2006.

[2] N. Barker, C. Myers, and H. Kuwahara. Learning genetic regulatory network connectivity from time series data. *IEEE Transactions on Computational Biology and Bioinformatics*, 2009.

[3] I. Cases and V. de Lorenzo. Genetically modified organisms for the environment: stories of success and failure and what we have learned from them. *International Microbiology*, 8:213–222, 2005.

[4] H. Kuwahara, C. Myers, M. Samoilov, N. Barker, and A. Arkin. Automated abstraction methodology for genetic regulatory networks. *Transactions on Computational Systems Biology*, VI:150–175, 2006.

[5] C. J. Myers. *Engineering Genetic Circuits*. Chapman & Hall/CRC, 2009.

[6] N. Nguyen, H. Kuwahara, C. Myers, and J. Keener. The design of a genetic Muller C-element. In *The 13th IEEE International Symposium on Asynchronous Circuits and Systems*, pages 95–104, March 2007.

[7] D.-K. Ro, E. M. Paradise, M. Ouellet, K. J. Fisher, K. L. Newman, J. M. Ndungu, K. A. Ho, R. A. Eachus, T. S. Ham, J. Kirby, M. C. Y. Chang, S. T. Withers, Y. Shiba, R. Sarpong, and J. D. Keasling. Production of the antimalarial drug percursor artemisinic acid in engineered yeast. *Nature*, 440:940–943, 2006.

Matthew William Lux[*] (mlux@vt.edu), Yizhi Cai[*] (cai@vt.edu), and Jean Peccoud[*] (peccoud@vt.edu)
[*]Virginia Bioinformatics Institute, Virginia Polytechnic Institute and State University

One of the essential keys to design automation is tools that allow designers to work at levels of abstraction. This abstraction requires well defined components with specific properties, careful rules for the assembly of components, a way to predict the output of an assembled system, and accessibility to the features of the tool. We have previously described the importance of such an abstraction in synthetic biology[1] and have developed a web-based tool called GenoCAD to fill this need. GenoCAD provides a library of carefully defined parts, guides users through the design of biologically reasonable constructs, and performs direct translation into mathematical models, all through a simple and intuitive web-based interface.

Users have different options for parts selection in GenoCAD. First, there is a carefully curated library of parts to choose from. We previously published an article on the development of biological registries[2] and a manuscript describing the specifics of this library is in preparation. The user can also access parts from the MIT Registry of Standard Biological Parts or create and store their own parts.

GenoCAD uses predefined syntactic grammars to guide users through the design process and output the designed sequence. The tool has preset rules on how parts are combined to prevent users from, for example, combining a promoter and coding sequence without a ribosome binding site (RBS). Details of how these rules are implemented have been published previously.[3] Currently, the main grammar is capable of handling the assembly of most published synthetic designs in *E. coli*, and is readily expanded. Alternative grammars for other organisms or applications are possible. For example, a grammar specific to the BioBrick standard is currently available with the tool.

Further, our group has developed software capable of translating a sequence composed of parts in the library directly to an SBML model. The software uses attributes assigned to parts and formal language theory from computer science to translate the designed sequence into a model. The software also allows for permuting any or all of the parts to create multiple model files for comparison of structurally identical designs. The specifics of the translation methodology are described in a paper under review and integration into GenoCAD is underway. Currently, the lack of association between parts and model parameters seriously diminishes the utility of such models, but assigning meaningful parameters to parts is a primary focus of our group.

Without accessibility, any such framework becomes useless. GenoCAD is web-based, has a professionally produced user interface, and from the beginning has been designed with the user in mind. A user guide was recently published in Nucleic Acids Research.[4] To improve practical usability, we are currently working on incorporating gene synthesis techniques[5] and lab automation into GenoCAD.

---

[1] Goler, J., Bramlett, B., & Peccoud, J. (2008). Genetic design: rising above the sequence. *Trends in Biotechnology*, *26*(10), 538-544.
[2] Peccoud, J., Blauvelt, M. F., Cai, Y., Cooper, K. L., Crasta, O., DeLalla, E. C., et al. (2008). Targeted development of registries of biological parts. *PLoS ONE*, *3*(7), e2671.
[3] Cai, Y., Hartnett, B., Gustafsson, C., & Peccoud, J. (2007). A syntactic model to design and verify synthetic genetic constructs derived from standard biological parts. *Bioinformatics (Oxford, England)*, *23*(20), 2760-7.
[4] Czar, M. J., Cai, Y., & Peccoud, J. (2009). Writing DNA with GenoCAD. *Nucleic acids research*.
[5] Czar, M. J., Anderson, J. C., Bader, J. S., & Peccoud, J. (2009). Gene synthesis demystified. *Trends in biotechnology*, *27*(2), 63-72.

Title: Bridging Synthetic Biology Design and Experiments using PoBoL

Authors:
Deepak Chandran[2*] <deepakc@u.washington.edu>
Michal Galdzicki[1*] <mgaldzic@u.washington.edu>
Alec Nielsen[2*] <alecn@u.washington.edu>

Affiliations:
[1]Biomedical & Health Informatics, University of Washington, Seattle, WA 98195, USA
[2]Bioengineering, University of Washington, Seattle, WA, 98195, USA
[*]Authors contributed equally

Abstract:
To facilitate exchange of information between computer applications used in synthetic biology we propose the use of a semantic model representation. The Provisional BioBricks Language (PoBoL) is a proposed electronic data exchange standard with the capability to serve this role (BioBricks Foundation Request For Comments 31). As synthetic biology employs diverse methodologies from computational modeling, information systems, and experimental techniques, there is a need for a common language to encode, store, and access data created by disjoint software tools. Representation of such elements using PoBoL will aid the efficient exchange of information between the three pillars of synthetic biology—computational models, genetic parts data repositories, and experimental procedures—and will pave the way for efficient design and engineering of biological circuits.

In a laboratory setting, use of PoBoL will facilitate synthetic biologists' ability to manage projects of increasing scope and sophistication with local parts libraries that can be searched and queried. The PoBoL format will also provide a universal framework for exporting, sharing, and publishing experimental data about genetic constructs. In addition, integration across physical parts libraries, computational modeling tools, and laboratory automation equipment will augment the capability for computer-aided design and automated synthesis of engineered biological systems.

In order to take advantage of generic software tools and common network infrastructure The Web Ontology Language (OWL) provides a W3C supported technology capable of expressing the complex description of the underlying data model. Use, adoption, and continued development of PoBoL to accomplish these ambitions should help the broader synthetic biology community.

Moving from computational models to the sequence of biological parts that represents the model can also be achieved using PoBoL. This transition can be demonstrated using a computer-aided design application called TinkerCell.

Towards a High-Level Programming Language for Standardizing and Automating Biology Protocols

Vaishnavi Ananthanarayanan and William Thies
Microsoft Research India
{t-vaisha, thies}@microsoft.com

For decades, biologists have relied on written descriptions of protocols to guide their experiments in the laboratory. However, due to recent technology trends, the practice of describing protocols with free-flowing English-language text is quickly becoming inadequate and obsolete. First, we are witnessing immense advances in laboratory automation systems. Microfluidic chips have been evolving at a pace faster than Moore's Law, with the number of valves per chip doubling every four months [1]. In order to leverage such technologies for biological experimentation, it will be necessary to express the protocols in a format that is not only comprehensible by humans, but also by machines. Second, the complexity of biology protocols is increasing dramatically. As we attempt to synthesize living systems as a composition of many parts, we will need to execute lengthy protocols with great precision. This will require a standard language for unambiguously describing the steps needed to synthesize a part, as well as for composing parts into a larger system.

We are developing a high-level programming language that enables standardization and automation of biology protocols. Our vision is to change the way that experimental methods are communicated: rather than publishing a written account of the protocols used, researchers will simply publish the code. As microfluidic devices mature, one could automatically replicate a colleague's experiment – or call it as a subroutine – by downloading the code to your own microfluidic chip. In the meantime, the code can be automatically converted to human-readable steps for manual execution in the laboratory. When written as a computer program, biology protocols can be parameterized to facilitate reuse in different contexts. They can also be mapped automatically to the setup of a given laboratory, taking into account the equipment and reagents that are available.

As a step towards this vision, we are defining and implementing the BioStream language for molecular biology protocols. In previous work, we demonstrated that an early version of BioStream could describe simple protocols and automatically execute them on different microfluidic devices [2]. Our current focus is expanding the language capabilities to encompass a broad and useful range of protocols. To date, we have expressed 15 protocols in the language, with protocols sourced from textbooks, classes, and published papers; our benchmark suite totals 2300 non-comment, non-blank lines of code. In the process of formalizing these protocols, we discovered and fixed several errors in the original descriptions. We have also implemented a C library that converts BioStream code to a human-readable format. Using this format, scientists at the Indian Institute of Science have successfully completed a DNA extraction procedure using our auto-generated protocol as the only reference. To the best of our knowledge, this represents the first time that a high-level programming language has been used to direct the actions of an experimentalist in a laboratory.

In the future, we look forward to collaborating with the synthetic biology community to explore ways of further standardizing their protocols, and to leverage a programmed description for maximum benefit to both manual and automated experiments.

Acknowledgments. We thank Utpal Nath, Mansi Gupta, Subhashini Muralidharan, and Sushmita Swaminathan for providing an experimental validation of BioStream, and Eric Klavins for helpful conversations.

References

[1] J. W. Hong and S. R. Quake. Integrated Nanoliter Systems. *Nature BioTechnology*, 21(10), 2003.

[2] W. Thies, J. P. Urbanski, T. Thorsen, and S. Amarasinghe. Abstraction Layers for Scalable Microfluidic Biocomputing. *Natural Computing*, 2007.

BioBrick Open Language: A Keystone for Efficient Development and Communication of Standard Biological Parts

Cesar A. Rodriguez[1], Doug Densmore[2], Drew Endy[1]

Affiliations: [1] Department of Biological Engineering, Stanford University, [2] Department of Electrical Engineering and Computer Science, University of California, Berkeley

Contact Information: cesarr@stanford.edu, densmore@eecs.berkeley.edu, endy@stanford.edu

Purpose

Presently, biological engineers develop the DNA sequence of a standard biological part using an informal language (informal symbols, syntax, and grammar). With the growth in availability of standard biological parts and the increasing capacity for DNA synthesis, there is a need for a formal language (formal symbols, syntax, and grammar) that describes the DNA sequence of a standard biological part. Simply stated, a DNA assembly language is needed. We put forth the BioBrick Open Language (BOL) as a forward engineering DNA assembly language targeted for use by biological engineers. We believe that BOL can serve as a keystone in an arch that efficiently connects an idea in the mind of a biological engineer to a formal and computable description of a standard biological part. BOL will be used to precisely describe an emerging design both textually and graphically (BioBricks Open Graphical Language, BOGL). BOL will be used to communicate designs between different software tools. For example, a design will be able to move efficiently between an integrated development environment (IDE), a laboratory management system, and a simulation tool. We hypothesize that BOL will increase the time efficiency in the development and communication of standard biological parts.

Methods

We elicited requirements for a DNA assembly language from approximately a dozen practicing biological engineers. We are developing an initial set of symbols, syntax, and grammar for a textual language, BOL, and a graphical one, BOGL. We are developing the language using a bottom-up approach starting with the simplest initial solution and increasing complexity in a iterative fashion. We have four biological engineers describing existing and developing new standard biological parts using BOL/BOGL. The emerging symbols, syntax, and grammar are posted on OpenWetWare where all interested parties are welcome to contribute to the development of the language. We are implementing a BOL/BOGL editor and BOL assembler for Clotho, a synthetic biology IDE and lab management system and BioBrick Studio Mobile, an experimental IDE on the iPhone/iPod Touch platform. Collaborators at University of Minnesota are implementing a BOL translator that will transform BOL code into a computational model that will run on SynBioSS, a modeling and simulation software suite.

Results

By August, 2009, we plan to have the first version of the BOL/BOGL specification, a BOL/ BOGL editor and assembler for Clotho and BioBrick Studio Mobile, and a BOL translator for SynBioSS.

Genetic Edge Detection

Jeff Tabor

Edge detection is a signal processing algorithm common in artificial intelligence and image recognition programs. We have constructed a genetically encoded edge detection algorithm that programs an isogenic community of *E.coli* to sense an image of light, communicate to identify the light-dark edges, and visually present the result of the computation. The algorithm is implemented using multiple genetic circuits. An engineered light sensor enables cells to distinguish between light and dark regions. In the dark, cells produce a diffusible chemical signal that diffuses into light regions. Genetic logic gates are used so that only cells that sense light and the diffusible signal produce a positive output. A mathematical model constructed from first principles and parameterized with experimental measurements of the component circuits predicts the performance of the complete program. Quantitatively accurate models will facilitate the engineering of more complex biological behaviors and inform bottom-up studies of natural genetic regulatory networks.

Authors: Timothy S. Ham, Zinovii Dmytriv, Paul D. Adams, Jay D. Keasling

Title: JBEI Registry: Towards a Distributed Web of Registries

Abstract:

As the desire for automation in synthetic biology grows, it is vital that a set of open, useful, and integrated toolkits are developed and made available. The Joint BioEnergy Institute (JBEI) Registry is an open source web application and platform that provides tools to organize, categorize and manage biological “parts” in a web accessible way. It is designed from the ground up to bridge the gap between “legacy” biological constructs and the new “BioBricks” paradigm by transparent support between plasmids, strains, and “parts”. Additionally, JBEIR has been designed for a distributed installation, so that different groups can run their own registries, and yet provide mechanisms for simple information exchange and data synchronization. It is built upon open source software, and licensed in the most liberal way to encourage adoption and participation. Or users can simply start using it from our web site.

Overcoming Abstraction Barriers in Synthetic Biology Systems

Austin Che
austin@csail.mit.edu

June 5, 2009

From my experience of writing the first version of the MIT parts registry to my continued involvement with the registry, I will present my perspective on some of the problems and possible solutions for the computer representation of synthetic biological systems. An abstraction hierarchy, which is often presented for synthetic biology, involves parts, devices, and systems. However, there are many practical issues to making such an abstraction hierarchy useful.

The parts registry implicitly assumes that there exists an abstract object known as a "part," but how do we define a biological part? In particular, we would like to allow for the automated exchange and reuse of information about a part across a variety of contexts. How should parts interact with different physical assembly standards? The result from assembling two parts together is commonly thought of as a new part, but how should a composite part be represented in a meaningful way across different assembly methods? I will present one possible way to specify physical assembly standards in a computer-readable form that allows us to easily represent a biological part.

"Devices" are often thought of as biological objects that can be functionally composed but there is currently no good instantiation of a registry of standard biological devices. Devices, to be usable, must have some physical instantiation, preferably in the form of parts. However, some valid device interconnects may not be physically possible due to the physical nature of DNA. Thus, any computer representation for devices should take into account both functional properties and the interface with previously defined parts.

Biological "systems" push the abstraction barrier even further. The heterogeneous nature of systems can pose even greater challenges for a potential computer representation. In addition, the boundary between devices and systems has not been well-defined. The practicality and usefulness of a parts, devices, and systems abstraction hierarchy depends on the resolution of many of these issues.

Title: PoBoL in the Lab: Structured Organization of Biological Samples for Laboratory Management and Automation

Authors:
Alec Nielsen[1] <alecn@u.washington.edu>
Deepak Chandran[1] <deepakc@u.washington.edu>
Michal Galdzicki[2] <mgaldzic@u.washington.edu>
Sean Sleight[1] <sleight@u.washington.edu>
Herbert M. Sauro[1] <hsauro@u.washington.edu>
Daniel L. Cook[2,3] <dcook@u.washington.edu>
John H. Gennari[2] <gennari@u.washington.edu>

Affiliations:
[1] Bioengineering, University of Washington, Seattle, WA, 98195, USA
[2]Biomedical & Health Informatics, University of Washington, Seattle, WA 98195, USA

Abstract:
As the ability to understand and reliably engineer biological systems advances, the increasing complexity, size, and number of engineered systems will prescribe the need for structured organization of biological samples. A standardized organization strategy will facilitate synthetic biologists' ability to manage projects of increasing scope and sophistication, and will also augment the spread of laboratory automation protocols. We propose the utilization of the Provisional BioBricks Language (PoBoL), a semantic description of both biological part functionality and physical manifestations of biological part samples (BioBricks Foundation Request For Comments 31). In a laboratory setting, use of PoBoL would aid the retrieval of biological samples from repositories and would support proliferation of standard automation protocols for high-throughput experimentation and development. Furthermore, because the PoBoL format links biological functionality with physical samples, other software tools could use PoBoL data to simulate and computationally model systems, and then automate the synthesis of these systems. Early examples of tools that could utilize PoBoL for biological sample tracking include the web-base registry BrickIt, and the CAD software Clotho. Automated management of these low-level laboratory and engineering details could help reduce the intractability of large-scale and very large-scale biological engineering. Adoption and development of PoBoL should support increased sharing of biological parts and collaboration within the synthetic biology community, more manageable laboratory organization for large-scale synthetic biology, and foster the spread of automation protocols for testing and developing engineered biological systems.

Title: TinkerCell, a CAD application for Synthetic Biology

Authors: Deepak Chandran, Frank T. Bergmann, Herbert M. Sauro

---

TinkerCell is a visual modeling application for synthetic biology with a C and Python interface that allows third-party algorithms or libraries to interact with TinkerCell's visual interface. At present, the functionalities that are added through the C and Python interface include deterministic and stochastic simulation C libraries, steady state analysis, flux balance analysis using LPsolve C library, graph analysis through the NetworkX python module, all the functionalities of PySCeS python module such as sensitivity and structural analysis, and a few unique functions, such as automatic generation of all combinations of a protein with multiple binding sites. The hope is for TinkerCell to serve as a host for various C and Python algorithms that the community has to offer. In addition to the C and Python interface, TinkerCell models contain meta-data pertaining to each item, which will allow direct communication with databases. The intent for such a structure is to support a biological parts database in the future. A parts database is one from which synthetic biologists can retrieve components for building synthetic networks, much like electrical engineers obtain electronic components. TinkerCell models can store information such as DNA sequences and data tables, which can then be used by the C and Python programs. TinkerCell's model framework supports modularity. Modules are constructed visually by placing a model inside a box and specifying the ``interface" for the module. Such modules can be connected using the interfaces, allowing complex models to be built using existing modules, thus allowing synthetic biologists to build new synthetic networks using existing ones. TinkerCell is an extensible and modular computer aided design application for synthetic biology presented in a simple graphical interface.

Toward Automatic Design of DNA Logic Gates and Devices

Ibuki Kawamata, Fumiaki Tanaka, and Masami Hagiya

Graduate School of Information Science and Technology, University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-8656, Japan
ibuki@is.s.u-tokyo.ac.jp, fumiaki@dna-comp.org, hagiya@is.s.u-tokyo.ac.jp

Recently, DNA logic gates [1] and DNA machines have been developed using only a simple complementary base pairing of DNA, that is, hybridization and branch migration. Because such reaction systems have been designed by trial and error, it has been difficult to design a complex system and to correctly verify the reaction. The purpose of our research is to develop a method for automatically searching and designing DNA logic gates [2] or DNA devices.

We propose a method that searches for a highly evaluated system from many candidates. The method increases the evaluation value gradually, which indicates how correctly the system works. We developed a computational DNA model and a simulator based on kinetic model to investigate DNA reactions and to verify the correctness. Though the searched system after the iteration may not be the global optimum, the system is a semi-optimal one, which has the highest evaluation value in the iteration.

Although there were several limitations, it was possible to design DNA logic gates such as AND, OR, NAND and NOR. The effectiveness of the proposed method is evaluated experimentally by a fluorescence experiment of an AND gate, which is designed automatically. It was also possible to simulate Shin's DNA walker [3] by our simulator.

It may be possible to apply our method to other devices such as DNA walker or DNA comparator where branch migration and hybridization are the center of the reaction. We plan to improve our system and design these devices automatically in the future.

References

1. Georg Seelig, David Soloveichik, David Yu Zhang, and Erik Winfree. Enzyme-free nucleic acid logic circuits. *Science*, 314(5805):1585–1588, Dec 2006.
2. Ibuki Kawamata, Fumiaki Tanaka, and Masami Hagiya. Automatic design of DNA logic gates based on kinetic simulation. *Proceedings of 15th International Meeting on DNA Computing (DNA15)*, 2009, to appear.
3. Jong-Shik Shin and Niles A Pierce. A synthetic DNA walker for molecular transport. *Journal of the American Chemical Society*, 126(35):10834–10835, Sep 2004.

Workflow Design for Synthetic Biology Using Clotho and Kepler

Douglas Densmore[1], Thien Nguyen[1], J. Christopher Anderson[2]
Department of Electrical Engineering and Computer Sciences[1]
Department of Bioengineering[2]
University of California, Berkeley
densmore@eecs.berkeley.edu, thien.nguyen@berkeley.edu, jcanderson@berkeley.edu

Two of the key requirements for introducing automation into the biological design process are reproducibility of specific protocols and formally capturing these protocols. Often these are very complicated, take considerable time to develop and "debug", and are lab/equipment specific. A highly modular, expressive, and extensible framework to capture and design these workflows would be tremendously useful. This work illustrates how the Kepler workflow design environment (a multi-university design effort) can be integrated into Clotho (a UC Berkeley based tool for the design of synthetic biological systems) to formally capture a number of specific design protocols related to composite biological part creation, composite part assembly, and part design/analysis.

Kepler (based on Ptolemy II) introduces the notion of an "actor based" workflow environment. Users connect specific actors under the control of a "director" to carry out a series of operations. Directors determine the semantics of the operations carried out by the individual actors. Example directors include: synchronous dataflow, process networks, and continuous time dynamics. Directors ultimately define how actors interact. Example actors include: database queries, file readers, file converters, I/O interfaces to external equipment, web service requests, MATLAB program interfaces, and arithmetic operations (just to name a few). Systems of actors can be composed into "composite actors" creating a hierarchical system where a system under the control of one global director may have composite actors which have collections of actors operating under the control of their own director.

The integration of Kepler with Clotho gives rise to a system which has database based design and visualization tools along with a rich workflow management framework. The functionality of Clotho becomes new actors for Kepler workflows. Protocols now become formally captured by the workflow and can be shared by all users of Clotho. Workflows can be specified in a lab agnostic way and then become lab specific when run locally through the lab's configuration of Clotho. Workflows can now be coupled with automation efforts as well (e.g. liquid handling robots).

This work provides several example workflows, their actual implementation in the lab environment, and the current progress on Kepler/Clotho integration.

For more on Clotho see: http://biocad-server.eecs.berkeley.edu/wiki/index.php/Tools

For more information on Kepler see: https://kepler-project.org/

Author Index

List of Particpants

Ananthanarayanan	Vaishnavi	Microsoft Research, India
Anderson	John	UC Berkeley
Bagh	Sangram	University of Toronto
Barmpoutis	Dionysios	California Institute of Technology
Caffrey	James	Life Technologies
Chappell	Gary	Life Technologies
Che	Austin	Ginkgo BioWorks
Clancy	Kevin	Life Technologies
da Costa	Bernardo	LS9, Inc.
Densmore	Douglas	UC Berkeley
Dmytriv	Zinovii	LBL
Endy	Drew	Stanford University
Galbi	Dave	Galbi Research
Galdzicki	Michal	University of Washington
Garcia Martin	Hector	Lawrence Berkeley Lab
Groban	Eli	LS9, Inc.
Gu	Chenjie	University of California at Berkeley
Gustafsson	Claes	DNA2.0
Ham	Timothy	JBEI
Hamada	Shogo	Tokyo Institute of Technology
Hassoun	Soha	Tufts University
Hillson	Nathan	JBEI/LBNL
Ho	Tsung-Yi	National Cheng Kung University
Holtz	William	UC Berkeley
Kawamata	Ibuki	Tokyo University
Kaznessis	Yiannis	University of Minnesota
Li	Peng	Texas A&M University
Liou	Jing-Jia	National Tsing Hua University
Lux	Matthew	Virginia Bioinformatics Institute
Madsen	Curtis	University of Utah
Marchisio	Mario	ETH Zurich, D-BSSE
McWalter	Ian	CMC Microsystems
Monzon	Eduardo	Utah State University
Mossawir	Katy	Verification Engineer
Murray	Richard	California Institute of Technology
Myers	Chris	University of Utah
Nicosia	Giuseppe	University of Catania
Nielsen	Alec	University of Washington
Papamichail	Dimitris	University of Miami
Peterson	Todd	Life Technologies
Reda	Sherief	Brown University
Riedel	Marc	University of Minnesota
Rodriguez	Cesar	Stanford University
Roychowdhury	Jaijeet	University of California, Berkeley
Salis	Howard	University of California San Francisco

Sauro	Herbert	University of Washington
Scheffer	Lou	HHMI
Shea	Adam	University of Minnesota
Steiner	Paul	MIT CSAIL
Tejeda	Abiezer	Utah State University
Thies	Bill	Microsoft Research India
Thomson	Ty	Plectix BioSystems, Inc.
Ullah	Ehasn	Tufts University
Voigt	Christopher	UCSF
Weiss	Ron	MIT - CSAIL
Winstead	Chris	Utah State University