Summary Report for the SemiSynBio Workshop on an EDA/BDA Interaction Roadmap

Workshop Dates:	August 19-20, 2016
Workshop Location:	Newcastle University, Newcastle upon Tyne, UK
Sponsors:	Semiconductor Research Corporation National Institute of Standards and Technology Newcastle University
Prepared By:	Jacob Beal and Brian Bramlett

Workshop Organizing Committee

Brian Bramlett / Intel (Chair)

Jacob Beal / Raytheon BBN Technologies

Doug Densmore / Boston University

Eric Klavins / University of Washington

Natalio Krasnogor / Newcastle University

Qinghuang Lin / IBM

Valeriy Sukharev / Mentor Graphics

Anil Wipat / Newcastle University

Victor Zhirnov / Semiconductor Research Corp.

Introduction

This report summarizes the discussions and findings from the Workshop on an EDA/BDA Interaction Roadmap that was held at Newcastle University in Newcastle upon Tyne, UK on August 19 - 20, 2016. The workshop featured contributions from a selected group of experts who provided perspectives on opportunities and challenges for development of biological design automation software and its potential for interaction with existing EDA tools and frameworks. The agenda and presentation materials can be found at https://www.src.org/calendar/e006098/ (log-in required). An ancillary goal of the workshop was to provide input information for the SemiSynBio Roadmap and to serve as background for decision-makers in industry, government and academia who are developing a vision for future EDA and BDA development.

The first session focused on the state of biological design automation (BDA) tools for design of biological systems and the relationship of these tools to existing EDA tools. In contrast with modern Electronic Design Automation (EDA), at present Biological Design Automation (BDA) is much more fragmented and task-specific. While "low-level" BDA that addresses the fabrication and optimization of nucleic-acid sequences is evolving rapidly, "high-level" BDA that focuses on system design appears unlikely to see major commercial growth until there are significant improvements in the information available about biological devices and the cost and time to build and test biological systems, particularly in the characterization and standardization of biological devices to support BDA. There is a great deal of similarity between EDA and BDA, however, and as these foundations are developed, there is a great potential to recapitulate a similar trajectory of success in design automation, and EDA tools and/or principles are likely to be one of the keys for doing so.

The second session explored the development of hybrid semiconductor/biological technologies and design automation. From a technology platform perspective, much of the focus was on microfluidics, and in particular, technologies utilizing semiconductor manufacturing techniques, compatible/scalable processes, and integrated electronic control and measurement. Implications for platform-level architectural approaches are becoming clearer (e.g., separation of wet and dry interfaces). As for design automation, adapting tools from EDA looks practical, with fairly direct reuse of many classes of tools; there does not appear to be a technical barrier so much as a lack a familiarity with EDA tool capabilities and the tendency for researchers to develop new tools for a specific purpose rather than deal with a learning curve to adapt. Finally, recent interest in using DNA as a digital storage medium shows great potential for driving the development of foundational new technologies integrating semiconductor and biochemical components.

The third session's focus was on lessons that can be derived from biology and utilized to improve semiconductor technology. This is a very challenging problem for the following reason: although we have gained substantial knowledge of the world of cells, metabolic reactions, neuro architecture, etc., we still need to know much more before we can effectively borrow from the biological world into the semiconductor world. Semiconductor systems are complex in the aggregate but fairly simple in their building blocks. On the other hand, the fundamental components of life, cells, are very complex structures with each cell performing remarkable computations. The lessons we will learn from biology might not simply help semiconductors but could theoretically have the potential to disrupt and displace the existing semiconductor ecosystem. However, there are limited learning opportunities that we can

benefit from in the short to medium term including, but not limited to: utilizing microorganisms for sensing applications; algorithms for processing information such as sound; neural architectures for lower energy; manufacturing through self assembly; and DNA-based memory.

The final session focused on what methodologies and design principles might support “bioprogramming languages” that can effectively handle the complexity of multi-scale electronic-biological systems integration. The discussion included theoretical foundations, design methodology and standards, research targets aimed at development of new engines for transformation and integration of synthesis artifacts, and effective methods for programmer interaction and feedback. Software design automation and representation are already becoming quite important for design of biological systems, particularly with regards to standards for design representation and data exchange. At the same time, biological data is expanding rapidly in volume and scope while suffering major problems in curation. The key for progress in this area appears to be a combination of standards, machine learning and inference technologies to assist with curation, and automation that can enable the adoption of effective design and test strategies by reducing the requirement for expensive and slow human-centric laboratory work.

In sum, the four sessions indicate high potential for development of valuable BDA tools, as well as for application and adaptation of existing EDA tools and principles in BDA. For the most part, however, this potential will require additional development of certain key underlying technologies before BDA can begin to have wide-scale commercial significance. In particular, strategic investment in characterization and standardization of biological devices, microfluidics-based engineering workflows, and curation and exchange of biological engineering data, are each likely to have a transformational impact on BDA within a 5-10 year time-scale.

EDA-BDA Synergy (Session 1)

Facilitator:

Anil Wipat, Newcastle University

Contributors:

Morgan Madec, University of Strasbourg
Jacob Beal, Raytheon BBN Technologies
Douglas Densmore, Boston University
Paul Bogdan, University of Southern California
Peter Carr, MIT Lincoln Laboratory

In contrast with modern Electronic Design Automation (EDA), at present Biological Design Automation (BDA) is much more fragmented and task-specific. BDA appears likely to remain so until there are significant improvements in the information available about biological devices and the cost and time to build and test biological systems. In the session keynote, Morgan Madec presented an overview of synergies and differences between EDA and BDA and discussed various ways in which existing EDA tools have been or are being adapted for use in BDA. Once of the major messages was that we have not yet achieved the necessary level of characterization and standardization of biological devices to support BDA. Jacob Beal focused on the challenges of characterization and abstraction, showing how these can simplify the engineering of previously fragile and tightly coupled systems. Douglas Densmore discussed the need for a better framework for interconnection, validation, and measurement of BDA, observing that many people are currently building BDA tools in isolation, leading to much duplicate and wasted effort. Paul Bogdan showed that one of the key limitations of BDA at present is the cost (in time and resources) of building and testing systems and the difficulty in accessing well-curated and relevant biological data, arguing a need to move toward verifying higher-level abstract behaviors rather than micro-level actions and values. Finally, Peter Carr discussed how microfluidics can miniaturize and accelerate building and testing systems, showing how EDA can be applied to raise the level of abstraction on microfluidics and related protocols and instrumentation to make them much faster and easier to apply, making fast and cheap biological build and test widely accessible.

State-of-the-art and current challenges

The value of design automation is determined primarily by the complexity of the designs to which it is being applied, and becomes high value only when the scale of the systems being engineered are well above the ability of small teams of humans to manage without computational assistance. One of the key questions in examining BDA is thus how to meaningfully define the complexity of the system that is being engineered.

In EDA, a useful yardstick for design complexity has been the number of transistors that are used in a design, which has generally risen exponentially over time following Moore's Law (Fig. 1). Similar exponential curves, constructed by Carlson [Carlson10], show a rapid increase in the number of DNA base-pairs in synthetic DNA constructs, recently reaching scales of 10^6, with synthesis toward a refactored *E. coli* [Ostrov16] and continuing to rise with the synthetic yeast project [Richardson17]. From this perspective, synthetic biology has clearly reached a point where it is at least valuable to have

"low-level" BDA tools for managing the optimization, editing, and assembly of large nucleic acid sequences - at the level of their fabrication as sequences of individual nucleic acids.

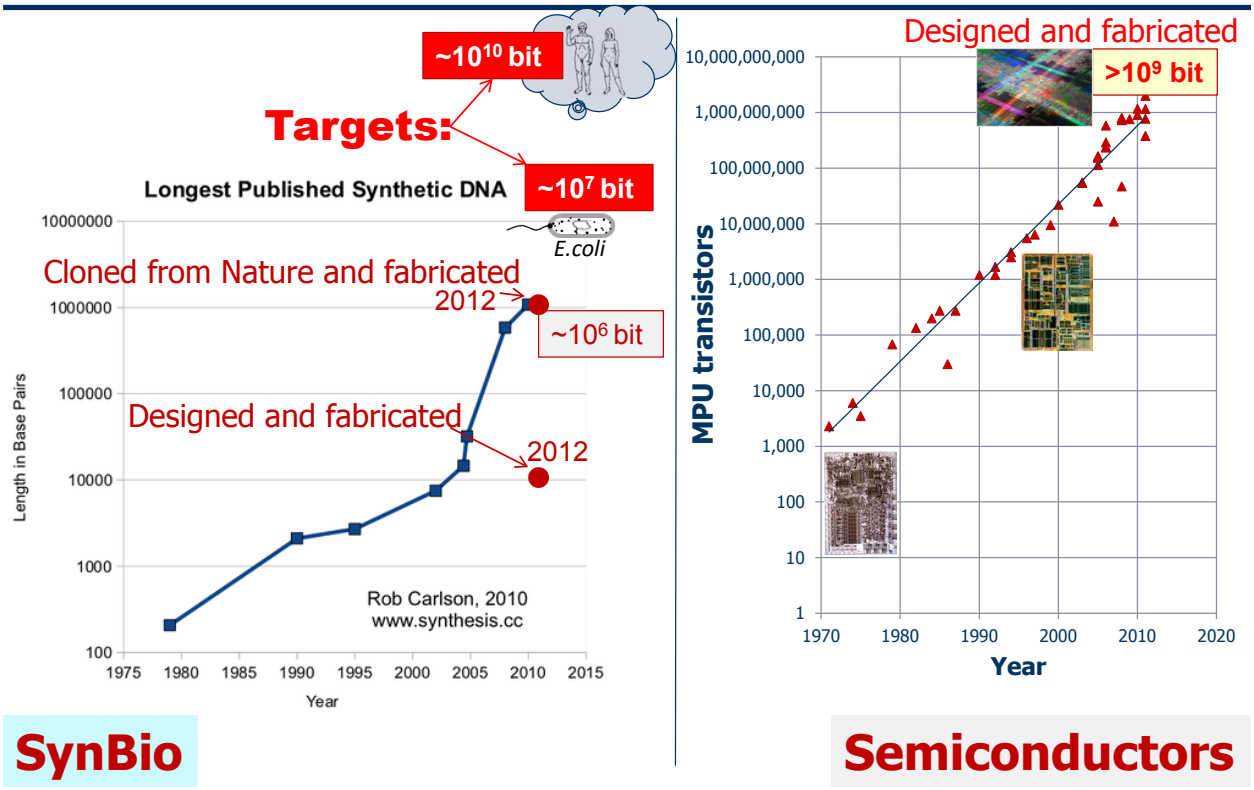


Figure 1: The number of nucleic-acid base pairs in fabricated biological systems has been rising exponentially, beginning to approach the number of transistors in semiconductor systems. This scale drives a rising need for low-level BDA [Zhirnov16].

It is important to realize, however, that base-pairs of nucleic acids are not a direct analogy to transistors. An individual transistor is a *functional* device, which is used to manipulate signals such as current flow and voltage level. An individual nucleic acid base pair, however, is not functional but rather a fundamental *structural* element. Nucleic acid base pairs are thus more closely analogous to pixels in the photolithography masks that are used to fabricate a silicon design.

A better comparison to transistors would instead be functional biological elements, such as a template that catalyzes molecular production (e.g., protein coding sequence) or a site where such production is regulated (e.g., promoter). While no consensus has yet emerged on how to draw boundaries for such counts, it is clear that by such measures the complexity of the largest engineered systems to date appears to be on the order of dozens of elements rather than millions. As one might then expect, there appears to be much less demand at present for "high-level" BDA tools that assist in the selection, arrangement, and optimization of functional elements and interactions in biological designs.

Guided by this perspective, we can compare the evolution of abstractions in semiconductors versus biology in order to inform our understanding of the likely preconditions to enable high-level BDA tools and the likely drivers for their demand. In semiconductors, abstractions rose over time in stages, from the basic physical theory to isolation of regulatory components, then to circuits made of those elements,

to standardized components for the modular assembly of circuits, and finally from standardized components to a progression of BDA tools for managing ever-increasing circuit complexity.

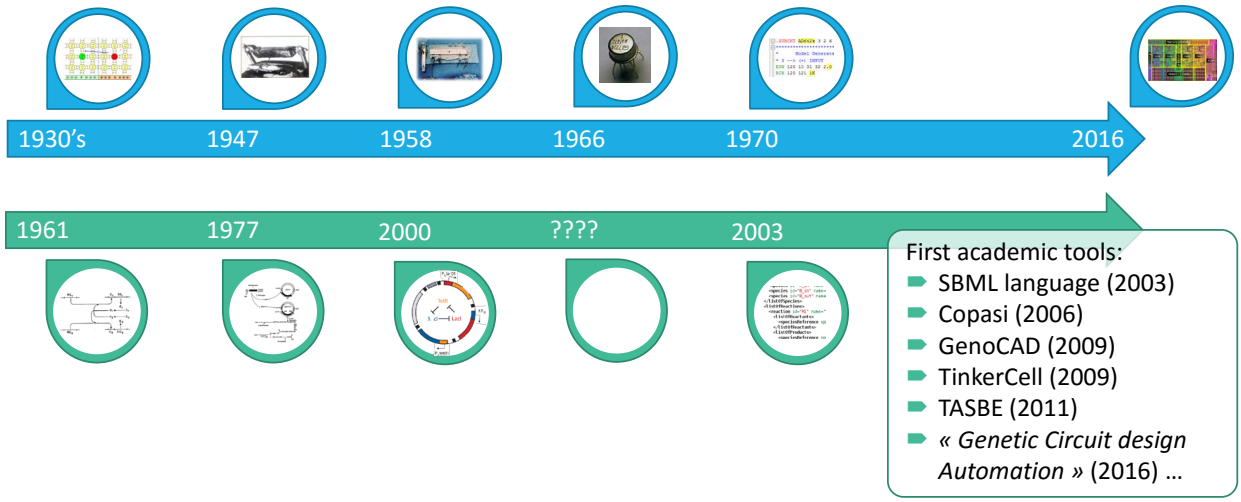


Figure 2: Semiconductor computing rose over time in abstraction, from physical theory, to regulatory component, circuit, standardized component, CAD tools, and finally exponential complexity growth. Biological computation abstractions are rising as well, but emerging BDA tools are not yet grounded in standardized components [Madec16].

For biological systems based on genetic expression, the basic underlying theory is well established, the components are identified and circuits can be constructed. Anticipating the BDA stage, many experimental tools have been constructed (Fig. 2). The utility and application of these tools, however, is currently significantly impeded by the lack of a sufficient system of standardized components for modular construction of circuits. Until this precondition is fulfilled, it is unlikely that modular circuit construction will proceed to the level of complexity that is necessary to support and drive development of an effective ecosystem of high-level BDA tools.

As noted, however, there is already both sufficient complexity and market to drive development of low-level BDA tools. In particular, these tools are predicated on the assumption that the functional aspects of a design have already been determined, and focus instead on the transductions from information to biological matter and back. For example, in converting from information to biology, BDA tools may be given a specification of a set of DNA sequences, then assist in the synthesis and assembly of those sequences, performing quality control on the products, and transforming those samples into the context where they will be evaluated (Fig. 3). Going in the opposite direction, BDA tools can assist in managing the execution of an experiment, applying instruments to measure performance, and collating performance data for interpretation. The value of BDA in this context is in allowing the engineers to focus more of their time and energy on the specification of the sequence (the "design" phase of a design-build-test loop), rather than the experimentation required for building and testing. Low-level BDA tools can also enable miniaturization and integration of the build and test processes, thereby allowing more processes to be run much more cheaply and possibly at a faster rate as well. There is already much work ongoing in this field, both in the academic and corporate worlds, with a particular emphasis on development of flexible hardware platforms (e.g., robotics and microfluidics), and associated supporting software. Some of the key technologies aimed at miniaturization (and how EDA may support them) were covered in Session 2; a separate branch of development aims at "cloud labs" that would allow outsourcing of build and test efforts.

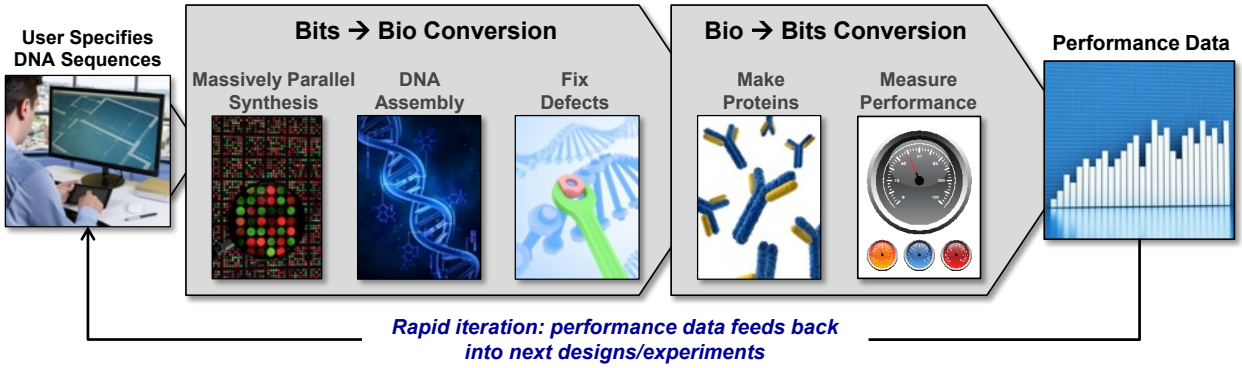


Figure 3: Low-level BDA focuses on automation of the transformations from information to biological matter and back, such as the exemplar tasks shown in this diagram [Carr16-1].

Returning to the question of high-level BDA, obtaining effective modular standardized components requires maturation of a number of different supporting technologies. In particular, in order to support an effective "design kit" for synthetic biology (Fig. 4):

1. Fabrication must be sufficiently reliable to allow designs to be realized with cost-effective yield.
2. Models must be available to predict the behavior of designs with sufficient precision to guide choices between competing design options.
3. Simulation tools must be able to evaluate those models in reasonable time.
4. Characterization procedures must be able to capture the information needed for models.
5. Libraries of characterized devices must share a standard description of this information.
6. Design rules must capture the intuitions of human experts for automatic application.

Only when all of these supporting technologies are available can effective design tools can be constructed to marshal them together into an effective "toolkit" for supporting biological engineering. Furthermore, note that these requirements are not tied to biology in particular, but are rather the general requirements for modularity that are encountered in any engineering discipline.

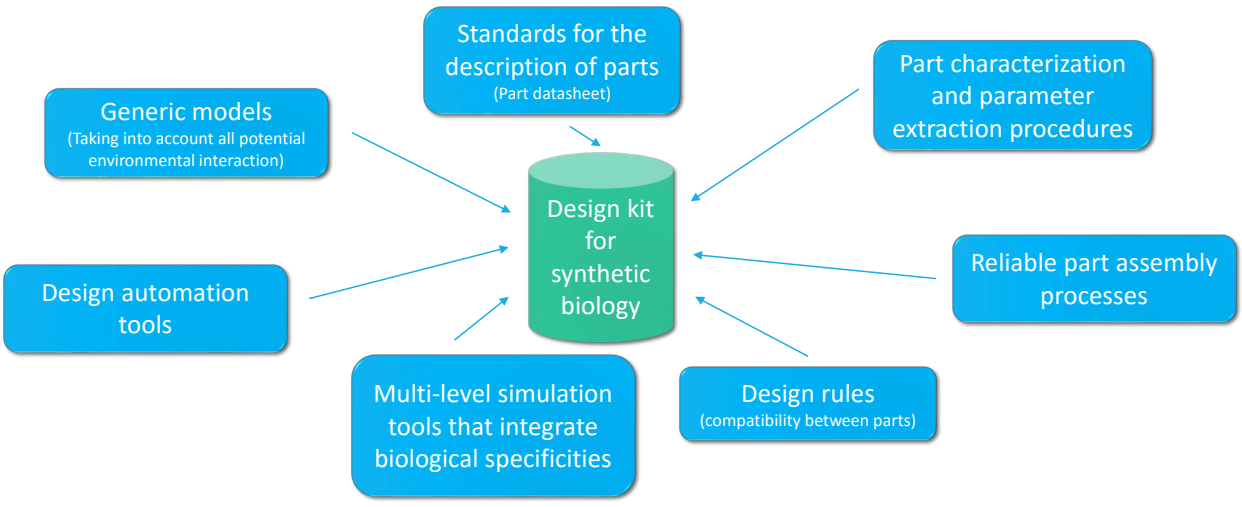


Figure 4: Supporting technologies needed for effective application of high-level BDA to organism engineering [Madec16].

Just as in the electronic world, a wide diversity of tools and approaches will likely be needed in order to cover the breadth of commercially significant organisms and engineering goals (Fig. 5). Success with only a small subset of organisms and goals, however, will likely be all that is needed in order to support development of a high-value commercial market.

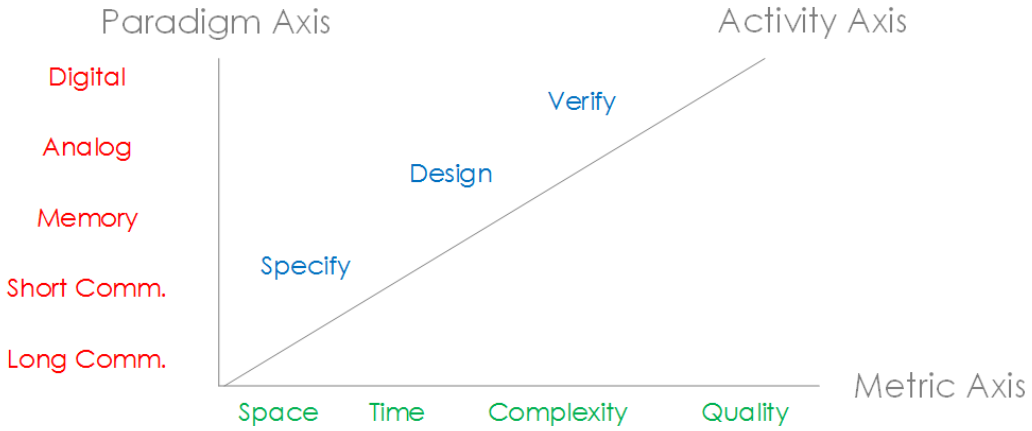


Figure 5: BDA tools may address many aspects of a complex space of opportunities [Densmore16].

Given the range of different scales and challenges involved in biology, it will often not be possible to address all aspects of a problem simultaneously in a single tool. Instead, it should be expected that a single problem will often need to be addressed by many different tools making different heuristic tradeoffs (e.g., speed vs. resolution vs. scale) and that these tools should be applied at different stages in an engineering workflow (Fig. 6). For example, [Wei13] reviews a number of cellular simulation tools that make different tradeoffs in scale and fidelity; all are applicable to significant engineering challenges and none dominates the others in value. BDA tools may also need to deal with the system itself changing on different scales of time, as an organism evolves in response to its environment. Genetic instability is not a certainty, however: many long-lived organisms are able to preserve their genetic stability quite effectively, and there is no reason to presume that it will be impossible for engineers to draw on similar principles to achieve stability in deployed systems as well.

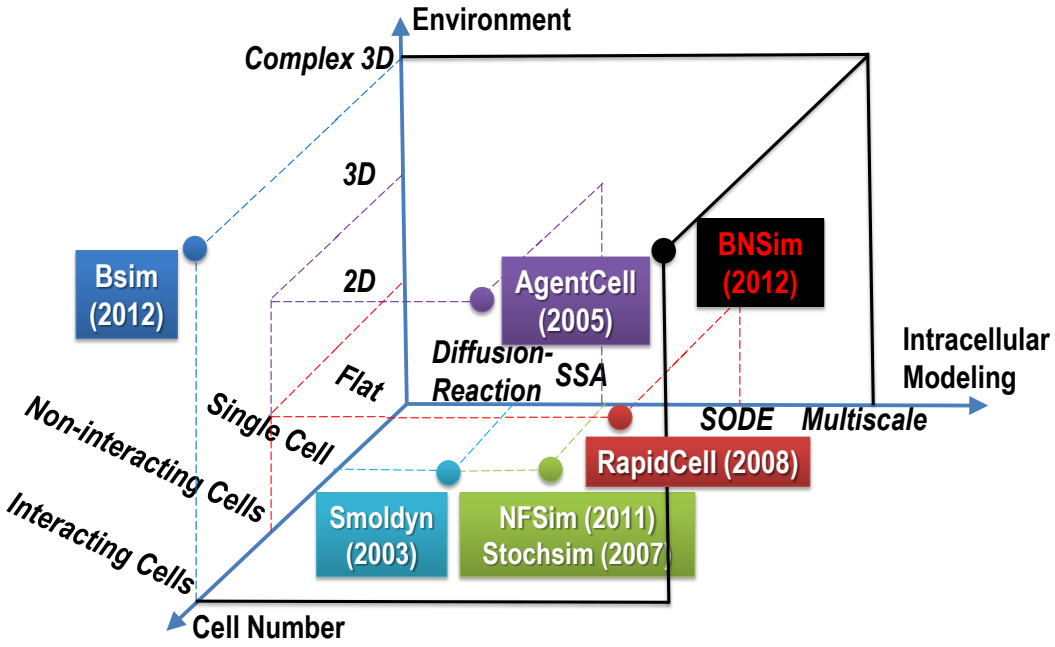


Figure 6: Differences in scale and fidelity requirements will lead to specialized niches for many different tools, as in this example of simulation tools from [Bogdan16].

In some cases there are specific opportunities for application of existing EDA tools. For example [Gendrault11] demonstrates that certain biological genetic regulatory network circuits can be mapped onto an equivalent electrical circuit description (Fig. 7). Once this mapping has been accomplished, then existing EDA tools for analysis of electrical circuits can be directly applied, and their results translated back to the biological circuit in order to predict its properties.

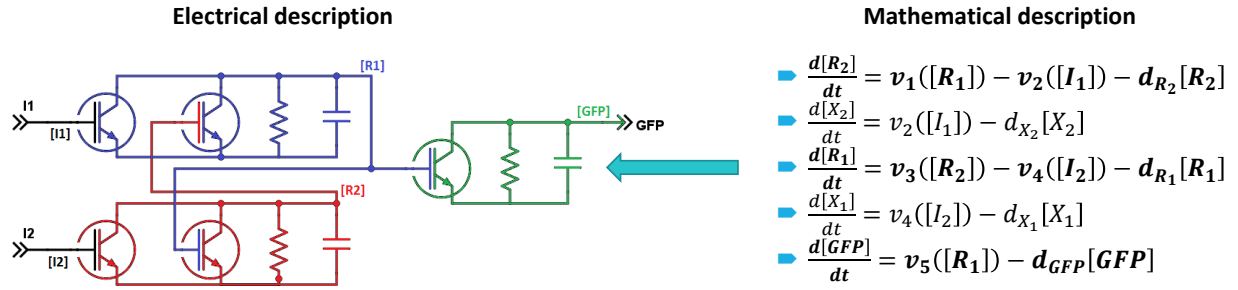


Figure 7: Example of direct EDA adaptation, for certain biological relations that can be modeled as equivalent circuits [Madec16].

In other cases, the particulars of biological engineering will impose requirements distinct enough that EDA tools are unlikely to be applicable, and BDA tools will need to be novel. This is already the case with many of the low-level BDA tools, which tend to be tightly linked with the biochemical particulars of biological processes and products. At higher levels of abstraction, it appears more likely that there will be more commonalities, driven by the universal and substrate-independent nature of information processing and control. The particular regions of that design trade-space that are emphasized, however, are likely to be different than are emphasized in much of EDA, e.g., involving more analog, hybrid, and uncertain elements.

As already noted, however, it will be difficult for high-level BDA to have a high degree of impact until the "component gap" is addressed. This will not be done by application of EDA/BDA tools, but knowledge of those tools and their supporting principles will be valuable for guiding the development of standardized components by providing development targets and metrics. Already, key advances have been made, such as the improved libraries of orthogonal devices (e.g., [Bonnet13], [Kiani14], [Stanton15], [Li15]), development of genetic elements that improve modularity in composition (e.g., [Mutalik13], [Lou12], [Carr16-2]), reproducible and comparable methods for characterizing device performance (e.g., [Beal15], [Beal16]), and model-driven methods for prediction of composite system behavior (e.g., [Davidsohn14], [Beal14], [Nielsen16]). These areas, and others, still require significant investment in order to progress, but there do not appear to be any fundamental challenges to progress, only a need for a large amount of research and development, coordinated and guided by EDA/BDA principles.

The synthetic biology community thus appears to be well positioned toward resolving the "component gap," if sufficient investment in research and development can be supplied. As progress continues in this area, the complexity of systems that can reasonably be contemplated for engineering should begin to rise sharply, and the need for high-level BDA tools will increase correspondingly. Ultimately, there will be a strong need to address many of the same sorts of complexity challenges that have been addressed in EDA, and technologies and experience from EDA will likely increase significantly in value in their application to BDA.

Finally, there are two less technical aspects of design automation are likely to be of high import in the development of BDA and where EDA may have critical contributions to make. First, as BDA makes it much easier for a potentially much larger population to engineer organisms, there are critical ethical and safety concerns that must be addressed. One of the ingredients in dealing with these considerations will be management of the security and traceability of designs. Design tools are important agents of monitoring and enforcement of standards in such areas, so techniques from the EDA community may be of use in this area. Second, one of the important services of EDA tools is to help in management of intellectual property, licensing, and contracting. Biological intellectual property practices, which have historically been focused on exploitation of individual elements, are not currently well organized to support componentization and composition, and EDA experience with both the organizational and legal aspects of IP composition as well as the tooling to support IP composition may be of high value in developing a mature biological engineering industry.

Session 1 Roundtable Discussion Summary

Discussions in this session centered around the following points:

- It is unclear whether enough is known about biology to effectively enable high-level BDA. On the one hand, experimentation can be used to get around poorly understood biological constraints, for example by exploring design space to find the solutions that do work. On the other hand, such exploration is not efficient and more biological knowledge will allow biological constraints to be more clearly defined and to be overcome via rational solutions rather than exploration. In practice, engineering is likely to progress along a spectrum ranging from brute-force exploration to rational design, but it is unclear where things currently stand or how quickly it is possible to progress.

- In general, biologists are not "ready" for BDA and are in fact still grappling with the lower levels of automation in the laboratory, e.g., LIMS. Most biologists have not yet been trained in any systematic and standardized approach to biological design. Applying more systematic and standardized approaches, however, does require design tools and is rapidly increasing in both prevalence and importance in biology.
- BDA is different enough from EDA that although much is shared the general principles, most existing EDA tools are unlikely to be directly applicable to BDA except in certain limited cases. Software design automation tools, on the other hand, operate at a greater remove from the computational substrate and are more likely to be directly applicable.
- Intellectual property is often a challenge in development of biological products: the intellectual property model used with EDA tools may provide a better means of dealing with managing these problems.

Forward Outlook

Analysis of the forward outlook for this topic is organized in several categories: (i) Major challenges and issues, (ii) Promising research topics, and (iii) Vision for the next 5 years, 10 years, and 15-20 years.

(i) Major challenges and issues

- Characterization and abstraction require better-curated data than most experimentalists are currently gathering.
- Need for identification and agreement on metrics for characterization of devices and modules that can support abstraction and decoupling of design elements.
- Multi-scale modeling, incorporating complex-systems understanding of self-organization, feedback, and emergent phenomena
- Need for clear and accessible metrics and benchmarks for success in BDA
- Experimental validation (or lack thereof) for designs needs to be incorporated as feedback into BDA tools and workflows.

(ii) Promising research directions/topics

- "Data sheet" standards for reusable, composable biological components, and improvement of available components so that there are large numbers of components that can be effectively reused and composed.
- Sequence "porting" tools to support transfer of components and designs from one host context to another.
- Sequence optimization tools, both for individual components and for dealing with the interfaces and interactions between components in a system.
- Tools to automate build and test of biological designs.
- Standards, practices, and workflow abstractions to support effective integration of BDA tools.
- Safety assessment and threat screening of designs.
- Development and integration of laboratory automation into routine biological engineering workflows.
- Adaptation of existing EDA tools for BDA purposes.
- Standards, practices, and tooling for handling of IP in biological designs.

(iii) Vision for the next 5 years, 10 years, and 15-20 years

5 years

- Widespread availability of effective and commercially viable BDA tools. Early tools are likely to focus on narrowing search spaces and elimination of bad solutions.
- Standards for characterization and composition of biological components, backed by large databases of useful components that conform to those standards.
- Standard interfaces and tools that enable flexible workflows customized to lab and project needs.
- Effective sequence porting and optimization and tools applicable to most common organisms and components.
- Widespread availability and integration of laboratory automation for key build and test workflow steps. Automation might be implemented either via local "black box devices" or via cloud/outsourcing.
- Integration of pathogen screening safety measures into key BDA tools.
- Effective exploitation of most EDA tools that are applicable to the BDA context.
- Routine BDA-assisted engineering of simple biological designs (< 10 functional units).

10 years

- Effective and commercially viable BDA tools based on analog and stochastic models.
- BDA workflows that enable complex designs largely without need for laboratory work, except for final verification and testing.
- Large numbers of "lab-less" biological engineers, similar to fabless electronics manufacturers.
- Mature informational and commercial ecosystem supplying biological components and modules at many levels of abstraction and complexity (by analogy to EDA, from "capacitor" to "op-amp" to "graphics card").
- Widespread availability and integration of laboratory automation for all workflow steps.
- Integration of generalized threat assessment and management into BDA workflows.
- Routine BDA-assisted engineering of complex biological designs (up to 100 functional units).

15-20 years

- Effective and commercially viable integrated BDA/EDA workflows for hybrid bio-electronic systems.
- Laboratory automation displaces most "by hand" experimental work.
- Mature BDA industry with segmentation of markets and separation of trades.
- Routine BDA-assisted engineering of biological designs at the scale of complex organisms (10^4 to 10^5 functional units)

Hybrid Semi/Bio technologies and Design Automation (Session 2)

Facilitator:

Jacob Beal, Raytheon BBN Technologies

Contributors:

Philip Brisk, University of California Riverside
Todd Thorsen, MIT
Andrew Hessel, Autodesk
Philip Gach, Sandia National Laboratories
Frans Widdershoven, NXP
Valeriy Sukharev, Mentor Graphics

Session Summary

This session explored the development of hybrid semi/bio technologies and design automation. From a technology platform perspective, much of the focus was on microfluidics, and in particular, technologies utilizing semiconductor manufacturing techniques, compatible/scalable processes, and integrated electronic control and measurement. Implications for platform-level architectural approaches are becoming clearer (e.g., separation of wet and dry interfaces). As for design automation, adapting tools from EDA looks practical, with fairly direct reuse of many classes of tools; there does not appear to be a technical barrier so much as a lack a familiarity with EDA tool capabilities and the tendency for researchers to develop new tools for a specific purpose rather than deal with a learning curve to adapt. Finally, recent interest in using DNA as a digital storage medium shows great potential for driving the development of foundational new technologies integrating semiconductor and biochemical components.

State-of-the-art and current challenges

To date, most instrumentation, and automation more generally, consists of discrete components connected together by purpose-built software, biochemical protocols, or manual implementation. The most advanced examples include the trend toward “lab in the cloud”, where some progress is being made on more abstract, parameterized descriptions of biochemical protocols and measurement. These typically room-sized platforms rely on robotics and automated fluidics systems to eliminate most human interaction, and are suitable for a certain scale of production.

In regards to characterizing the scale of such systems, it may be useful to look at major parameters such as working volumes and cell counts in addition to more conventional metrics such as device count and feature size.

Building microfluidic systems at scale is a challenge [Brisk16]. Semiconductor and electronic fabrication depend on scale (volume and integration) to drive costs down to pragmatic levels. Currently, however, most microfluidic chips are ASICs: universal programmable machines may be unlikely due to the diversity of sensing and biological operations. That said, there is an enormous role for low-cost prototyping, and modular, even standardized construction systems and component libraries are beginning to close important gaps between one-offs and mass production, such as the system shown in Fig. 8:

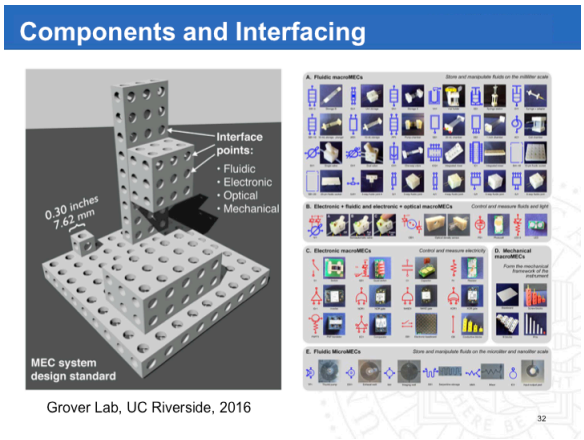


Figure 8: Example of a standardized library closing the gap between one-offs and mass production [Brisk16].

This example shows an integrated subsystem customized for a common biological fabrication flow. Similar devices have been fabricated on printed circuit boards, capable of handling single cells. The complexity and scale shown here approach limits for single-layer integration on glass without embedded active components. Integration into larger systems (microfluidics, microscopes, thermal control, cell counters, etc.) also limits the miniaturization and capacity/throughput. While this component is well-integrated, the larger system is not. Similar devices are fabricated using printed circuit board technologies, and have potential for integrating active components, even embedded silicon.

While most technologies currently used are not directly compatible with semiconductor processes, electrowetting shows some reuse of electronic manufacturing processes and materials (Fig. 9). In particular, electrowetting technologies are broadly compatible across different electronics manufacturing platforms, and show great potential as a core integration technology [Brisk16].

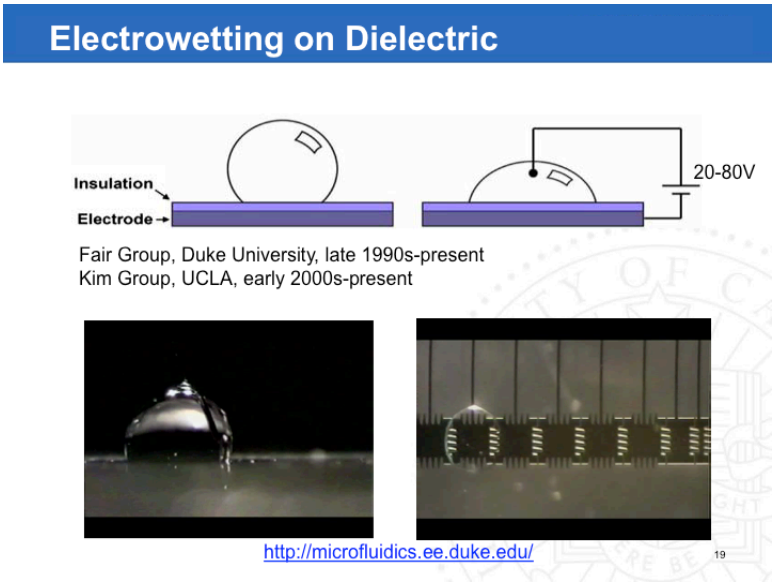


Figure 9: Electrowetting provides improved compatibility between biological and silicon surfaces [Brisk16].

Recently electrowetting platforms with increased integration density and functionality (e.g., embedded temperature control, optical feedback) have been demonstrated [Gach16]. Scaling up throughput and reducing cost per operation has been seen to bring a clear benefit.

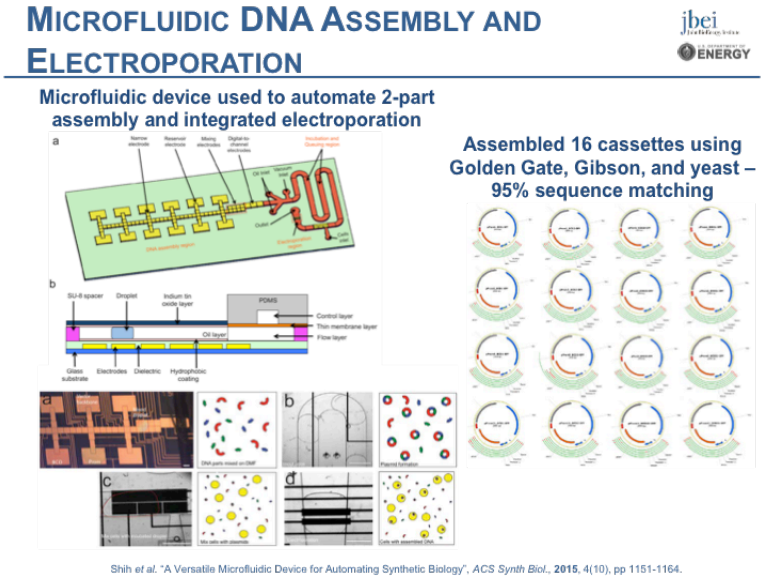


Figure 10: Example of integrated microfluidic system [Shih15, Gach16].

CMOS itself is clearly becoming the platform of choice for applications such as sequencing and lab-on-a-chip assay-scale measurement. CMOS feature sizes and sensitivities scale from the cellular to the molecular (Fig. 11), but the intermediate scale, system assembly and integration technologies are critical to effectively accessing the potential of CMOS-based technologies beyond architecturally simple sensor components [Widdershoven16]. The cellular and molecular scales also represent distinct but coupled design regimes, each requiring its own set of specific technologies. A key observation is that these two regimes should be developed on a mutually compatible technology platform. Some possible design conventions are beginning to emerge, such as segregation into a "wet" fluidics/optical side and a "dry" electronic and thermal side. Using standard interfaces is key to scalable "wet-dry" co-design [Widdershoven16].

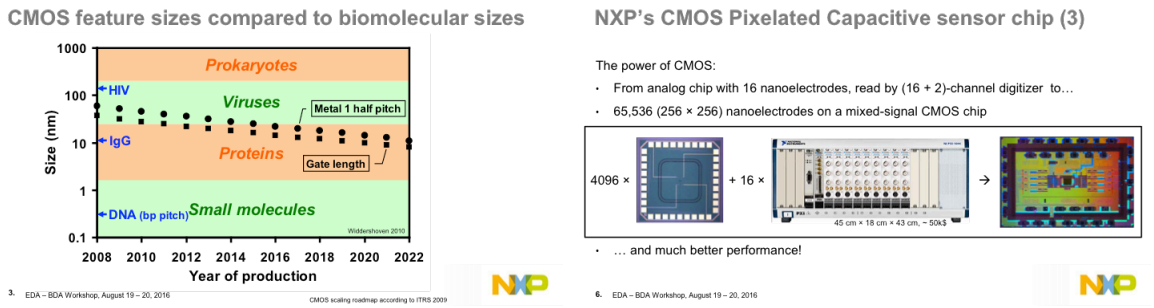


Figure 11: Scale similarities between silicon and biological components in CMOS [Widdershoven16].

Commodity technologies are being utilized, but it is difficult for the biological community to access more advanced technologies that have not been tailored to biological usages in high volume manufacturing. Integrating silicon with microfluidics and/or optical systems requires deviating from the main CMOS

development route. Because of the huge costs involved this appears likely to have a hard time getting financed. Although keeping the silicon, microfluidics and optical technologies strictly separated significantly limits the size of the solution space, in practice the impact of this separation appears unlikely to be severe because it is mainly the commercially or technically non-viable "solutions" that will be pruned. Ideally, a silicon-based platform (e.g., electrowetting, possibly with some extensions and enhancements) could be developed to generalize and subsume the capabilities of existing non-silicon-based platforms, which could then be tailored to biological usages in high volume manufacturing.

With regards to tools, EDA shows strong potential for being adapted for use in a BDA context while retaining complementary EDA function. For example, tools used for physical simulation and analysis of thermal design in packaging may be adapted to also model microfludic device physics (Fig. 12). Until system design reaches the level of requiring the capabilities of CAD frameworks to flexibly integrate tool sets across multiple design domains, however, it is unclear if adoption of EDA tools (with their inherent learning curves) would serve early-stage research and development.

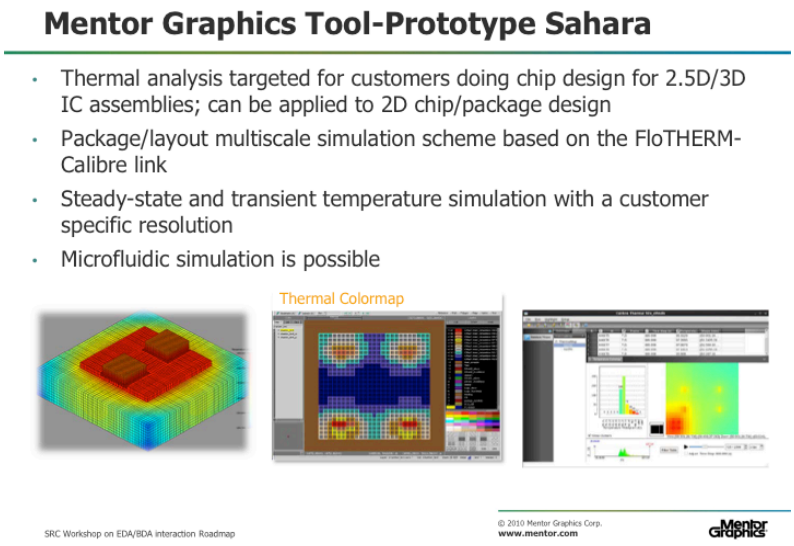


Figure 12: Example application of EDA tools to microfluidic designs [Sukharev16].

An example of domain-spanning design is the work presented in [Thorsen16], which described a layered approach to development of hybrid fluidic systems. This begins from protocol description languages (architecture-independent), and then proceeds to fluidic instruction sets before ultimately mapping to hardware primitives (Fig. 13). This system has permitted the design, construction, and operation of a feedback system for dynamically adjusting chemical inputs based on sensor data.

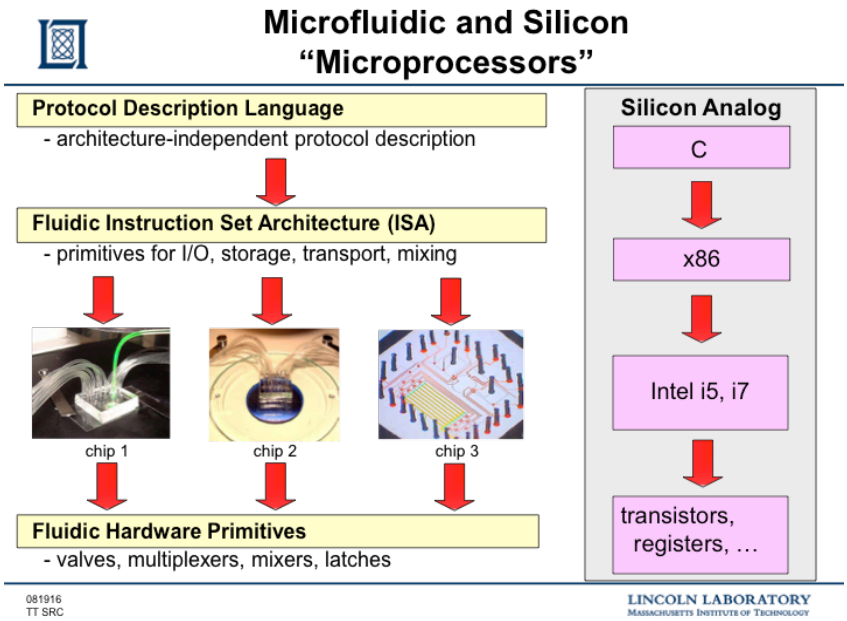


Figure 13: Example of integrated fluidic design flow [Thorsen16].

Finally, many of the areas discussed could be driven much more coherently with a coordinated agenda if a high-volume design of sufficient complexity was targeted. For example, today, DNA synthesis is a clear bottleneck for several grand applications of DNA technology, such as synthesis of whole genomes or using DNA for digital storage [Hessel16]. Semiconductor-based genetic synthesis needs to be explored in order to enable a next-generation DNA synthesis market. DNA digital storage may provide a target with sufficient economic payoff for the required investment, and this should benefit biological applications as well. Storing and manipulating large numbers of very small volumes of material is one potential approach to bringing such systems down to chip- or package-scale.

A second area with potential for pharmaceutical and other industrial biotech as a driver/collaborator are lab-on-a-chip platforms, especially ones capable of integrating *in situ* cellular culture and measurement with molecular sensing. The economic tradeoff here is lower volume than DNA memory (but still high volume manufacturing) for a higher-margin application.

In either case, a small number of key applications could provide the focus and structure needed to develop a coherent, scalable platform. Examples like these could form the basis of a roadmap spanning from more easily achievable devices (e.g., advanced lab-on-a-chip) to the more challenging ones (e.g., DNA memory).

Session 2 Roundtable Discussion Summary

One of the major barriers to adoption of microfluidics and other hybrid semiconductor/biological technologies is the complexity of engineering these devices, particularly since there are many different types of expertise needed. CAD tools per se are not the key barrier, but the larger EDA pipeline, including fabrication, build, and test automation. Many types of protocols and assays can be miniaturized this way, but there needs to be a significant degree of flexibility and receptivity to user customization in order to be applicable to a large number of practitioners. The device/world interface is also a problem, which might be addressed through standardization.

Forward Outlook

Analysis of the forward outlook for this topic is organized in several categories: (i) Major challenges and issues, (ii) Promising research topics, and (iii) Vision for the next 5 years, 10 years, and 15-20 years.

(i) Major challenges and issues

- Many issues are economic rather than technical; semiconductors are predicated on high-volume manufacturing, which in turn requires large markets for a given design.
- Lack of standardization, at least in terms of interfaces for integration/assembly.

(ii) Promising research directions/topics

- Application of EDA tools, workflows, and practices to microfluidics.
- Standardization of chip interfaces to reduce complexity of customization.
- Direct integration with CMOS sensing and actuation technologies.

(iii) Vision for the next 5 years, 10 years, and 15-20 years

5 years

- Accessible microfluidics for most common assays.
- Synthesis of the expertise of the biological and semiconductor communities, as well as experts in software development, to design hybrid bio/CMOS chips with applications in DNA synthesis, read/write nucleic acid-based memory.
- Exploration of new biocompatible materials that can be readily integrated into the fabrication path.
- Accessible software design of silicon-based microfluidic devices, similar in complexity to current FPGA or printed circuit board design.

10 years

- Desktop laboratory-in-a-box systems.
- Chip fabrication and realization of the designs developed in the 5 year vision.
- Automatic conversion of a high-level description of a biochemical reaction or ongoing biochemical process into a customized device that can efficiently execute the reaction and/or process.

15-20 years

- Cheap consumer laboratory-in-a-box.
- Truly hybrid bioCMOS chips, combining biological and electronic logic.
- Silicon-based DNA storage technology with CAD-designed fluidic subsystems.

New Design Principles for semiconductor systems inspired by biological systems (Session 3)

Facilitator:

Rafic Makki, GLOBALFOUNDRIES

Contributors:

Samuel Perli, MIT
Yong Zhang, Cadence Design Systems
Oshiorenoya Agabi, Koniku
Gregory Parsons, North Carolina State University
Alex Yakovlev, Newcastle University
Victor Zhirnov, Semiconductor Research Corporation
Natalio Krasnogor, Newcastle University

Session Summary

Although the progress of CMOS technology has been extraordinary, submicroscopic computers remain outside of our grasp. However, nature appears to have successfully addressed the submicroscopic design challenge, and may suggest new solutions for future microsystems for information processing. Deeper understanding of the principles of cellular information processing may enable new generations of computing systems. Among the most promising characteristics of biological computing is the extremely low requirement for energy of operation, much closer to thermodynamic limits than the semiconductor systems, which may address the semiconductor industry's grand challenges of *energy* and *cost*. The industry's innovation model that has been centered on Moore's law is changing. This community needs to find ways to make a more compelling argument for research funding and take advantage of this changing innovation model. This session's focus was on lessons that can be derived from synthetic biology and utilized to improve semiconductor technology. This is challenging for the following reason: although we have gained substantial knowledge in recent years into the world of cells, metabolic reactions, neuro-architecture, etc., we still need to know much more in order to attempt to realize the conjectured potential for biology-based disruption of today' computing architectures and other practices of the semiconductor information and communication technology (ICT) world.

Semiconductor systems are complex in the aggregate but fairly simple in their building blocks. On the other hand, the fundamental components of life, cells, are very complex worlds with each cell performing remarkable computations. The lessons learned from biology may not just influence the evolution of semiconductor technology but might actually significantly disrupt the $350 billion semiconductor ecosystem, which is expected to grow at a CAGR of 6.7% over the next ten years.

There are some learning opportunities that we can benefit from in the short to medium term including, but not limited to: utilizing microorganisms for sensing applications; biology-inspired algorithms for processing information such as sound, neural architectures for lower energy, manufacturing through self assembly, energy-modulated design where information flow would be commensurate to energy flow, and DNA-based memory.

State-of-the-art and current challenges

For this session, one of the clearest takeaways is that knowledge of biological systems is nascent, especially in terms of both the scale of systems which can be predictably engineered, and in using that knowledge to drive new approaches in semiconductors and computing.

Little of known advantages of biological systems, including low power, high volume of potential information processing capabilities, have been realized in technologies in computing systems. Application of the qualities of either cells or brain architecture to semiconductor designs is limited by available data and knowledge about biological systems. One focus may be to develop computational methods to analyze biological data that are available but rarely being used (e.g., the rapidly accumulating databases of sequence and microarray data), in order to gain better understanding of biological phenomena/behaviors, and to build better models that can be improved by comparing simulation results and real data.

New abstractions may result from developing new mathematical tools or borrowing analytical tools from statistical physics and quantum mechanics to mine biological data (potentially with new machine learning techniques for uncovering and managing interactions in synthetic biology) and provide a system level understanding. These models might in turn form the basis for developing new abstractions and programming models for both biological systems and bio-inspired engineering approaches.

In principle, living cells can be used for computing, as biological systems implement very low energy, yet complex information processing, while interfacing with many different modes of inputs and outputs spanning chemical, electrical, temperature, pressure and optical signals [Perli16]. Figure 14 provides a comparison between Si-based "cells" and biological cells. It shows that in today's technologies, a Si-based cell cannot match a biological cell in operational energy [Zhirnov16].

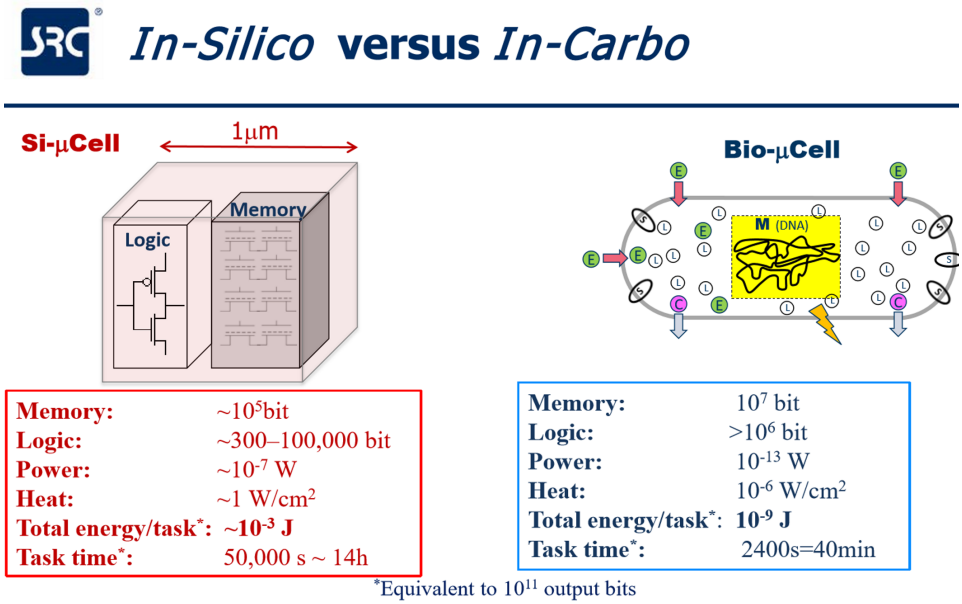


Figure 14: Theoretical logic gate equivalence in living cells [Zhirnov16].

Learning from neural cells might open new opportunities for extremely low power systems, but in order to fully realize the potential of such systems, it appears likely that new simulation and design tools

would have to be developed, including quantitative models for neural simulation that are likely to be stochastic and have no direct EDA equivalent [Zhang16]. Hybrid computing systems, whether neural or otherwise, would need a well-defined interface between biological and electronic layers of implementation, requiring standardization and simulation of biological systems from receptors to whole cell interaction [Agabi16].

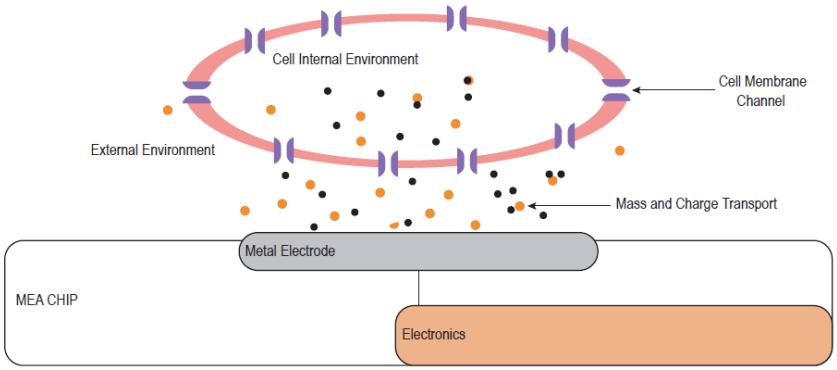


Figure 15: Conceptual design for a generic hybrid information processor utilizing an interface between biological and electronic layers of implementation [Agabi16].

Figures 16 and 17 give some examples of how living cells can model digital and analog operation [Perli16]. In terms of efficiency, a good place to start is co-localizing computation and storage, as in the example digital counter that follows. To date, tight coupling of computing and memory has not been common in silicon architectures.

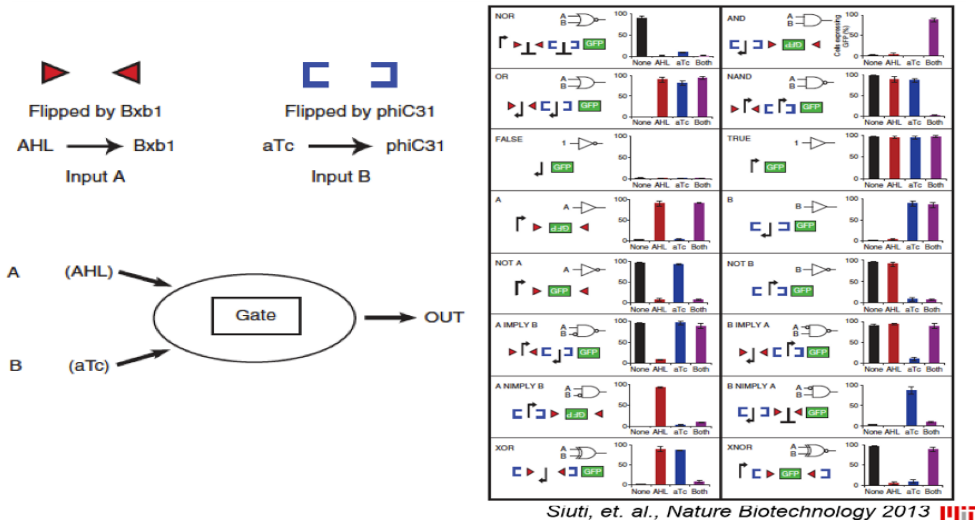


Figure 16: Logic gate equivalence in living cells [Perli16].

Number of inputs	Register design	Number of states
1		2
2		5
3		16
4		65
5		326
6		1957
7		13700

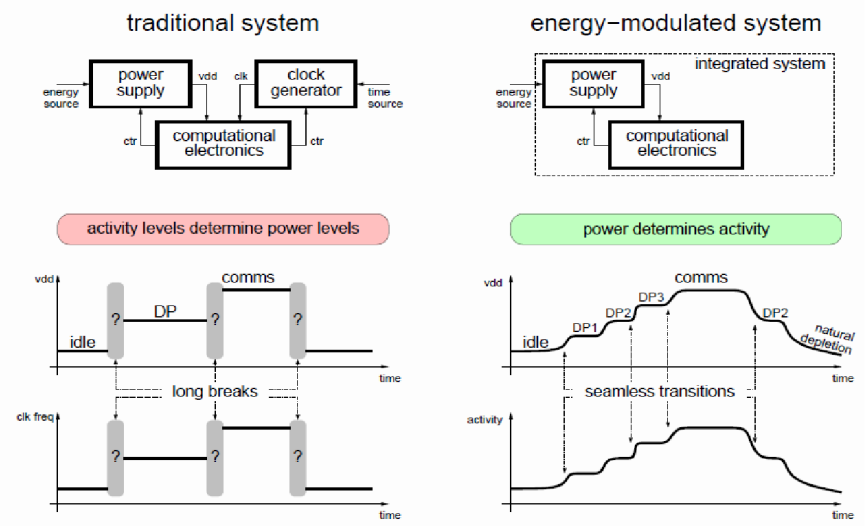
Figure 17: Modeling sequential machines in living cells [Roquet16]

In order to support computation with living cellular systems, BDA needs a clear definition of device function and performance [Parsons16]. There is still an open question, however, on what are useful irreducible device elements for engineering computation in cellular systems. There is also a need for understanding more about how a hybrid bio-electronic system might benefit from stochastic events, and a need to further develop methods for electronically accessing and controlling chemical activity, including electrically addressed gene expression [Parsons16].

Automated design (e.g. based on machine learning) may also offer new opportunities for EDA-BDA interfaces, both in the understanding of underlying biological-based substrates and providing new design principles, ultimately increasing automation in the programming of living matter [Krasnogor16-1].

Quantifying computation in living cells versus silicon is a difficult problem. Most of the literature is strongly tied to digital systems and a von Neumann style of conceptualization of computing. However, it might be more useful to look at organism-level computation instead of living cells. What does the organism compute at an instantaneous time and how many resources does it deploy for that function? Can those resources be cloned, especially using a similar substrate (i.e., a biological tissue or wetware)? Furthermore, much processing in biological systems stems from the multimodal, plastic nature of biological systems. For example, this allows a single neuron to be massively non-linear, and to potentially perform many functions depending on context [Yakovlev16].

Another concern is the possibility of switching from a performance-driven energy-efficient paradigm to energy-driven and space-limited performance-optimized systems design, in which computation should be quantified in decisions per Joule. Figure 18 shows a comparison between legacy systems and energy modulated systems [Yakovlev16]. A key challenge here for semiconductor systems is how to achieve massive informational connectivity at all levels of hierarchy or spatial layers of powering and timing.



4

Figure 18: Traditional digital vs. energy-modulated view [Yakovlev16].

Session 3 Roundtable Discussion Summary

The roundtable discussion for this session centered on the following questions and related answers:

- How can this community pool its intellectual resources and make the right case to attract the right level of funding from the semiconductor industry?
  Funding in areas directly related to the topic of the session is scarce. It was pointed out during the discussions that recent efforts such as the SIA-SRC report on "Rebooting the IT Revolution" make a good case for additional government funding. These kinds of industry-driven results can make a significant difference in attracting funding.

- To what extent can lessons from biology be applied in semiconductors and is enough understood about cell computation and metabolic processes to extract lessons and use them in the right context for advancing semiconductor technology?
  Biological systems make complex decisions with great energy efficiency. However, semiconductor systems process algorithmic type operations much faster than humans can. It would be a tremendous leap forward if semiconductor technology could also process information and weigh tradeoffs like a human brain can (neuromorphic design). However, as noted earlier, there is still much to be learned about biological systems before we can do similar things in semiconductor technology.

- Would these lessons disrupt the industry and create a new value system?
  If biological principles turn out to be fundamentally different, lessons from biological systems would have a disruptive impact on semiconductor technology.

- How should today's semiconductor industry prepare for such disruption?
  The industry can prepare by investing in R&D programs in partnership with the synthetic biology community, both academic and industrial. This is consistent with the semiconductor industry's

push towards MtM technologies, especially as we approach the end of device shrinking in the next 7-10 years.

- To what extent is the analogy between semiconductors and synthetic biology valid?
  Attempts have been made to establish analogies, at a certain level of abstraction, such as the similarities between chemical reactions induced by enzymes and the operations of a logic gate. These analogies are valid at the right level of abstraction. There are also valid analogies between DNA memory and semiconductor memory that can be made.

- Biosystems do more computations and operate on much lower energy – how can we incorporate this into semiconductor technology?
  There are two factors to the energy issue: fundamental operation and architecture. Semiconductor systems push electrons to communicate, but in biological systems such as the brain, neural excitability is driven by chemical (ion concentration) activity and electrical activity. At the architecture level, semiconductor systems are mostly von-Neumann-type architectures regulated by synchronous clocks, whereas what we know so far about the brain is that the architecture is fundamentally different, with an average of 7000 synapses per neuron resulting in over 1000 trillion synapses by some estimates, providing asynchronous communication among neurons. Achieving the energy efficiency of the brain will likely require fundamental changes in scale, architecture and physical operation.

Forward Outlook

Analysis of the forward outlook for this topic is organized in several categories: (i) Major challenges and issues, (ii) Promising research topics, and (iii) Vision for the next 5 years, 10 years, and 15-20 years.

(i) Major challenges and issues

- Lack of sufficient depth of understanding of many details of biological systems.
- Definitions of biological computational potential not yet well settled.

(ii) Promising research directions/topics

- Biological principles for information processing in analog and noisy environments.
- Blurring the distinction between computation and memory.
- Biology-inspired low energy computation.

(iii) Vision for the next 5 years, 10 years, and 15-20 years

5 years

- First market-ready sensor systems based on biological tissues.
- Increased understanding of computational potential and nature of biological systems.

10 years

- Neural computation understood well for at least some brain areas or functions.
- Some commercial biologically-inspired computing systems.

- Biological principles exploited for order-of-magnitude reduction in energy budget required for information processing.

15-20 years

- Application-specific biologically-inspired computing systems become readily available.
- Computational devices based on neurons commonly used for specialized applications enabled by form factor and power consumption.

Software Design Automation for Complex Biological and Electronic Issues (Session 4)

Facilitator:

Andrew Hessel, Autodesk

Contributors:

Mike Holcombe, University of Sheffield
Chris Myers, University of Utah
Sumit Jha, University of Central Florida
Anil Wipat, Newcastle University
Natalio Krasnogor, Newcastle University

Understanding and engineering biological systems sets new goals and challenges for software engineering. Is it possible to devise new methodologies and design principles that embrace (rather than shy away from) the complexity of multi-scaled electronic-biological systems integration? The focus of this session was thus to prospect for "bioprogramming languages" and design representation standards that embrace multi-scale processes, automated program synthesis tools to create software that meets specifications for complex biological-electronic systems, etc. The discussion scope included theoretical foundations, design methodology and standards, and research targets aiming development of new engines for transformation and integration of synthesis artifacts, and effective methods for programmer interaction and feedback. The keynote for this session was given by Mike Holcombe, who argued that current formal verification approaches are likely of limited use for biological systems due to the complex systems nature of both organism and environment, and then presented the X-machine formalism as a possible candidate approach to specification. Sumit Jha took an opposing perspective, arguing that well-curated models and experimental data will allow formal verification and synthesis to apply effectively to biological systems. Chris Myers showed that lower-level genetic design aspects of biological design are already benefiting from automation and how standards are key to enabling abstraction, decoupling, and interchange. Anil Wipat observed that there is already a vast amount of available biological data that is barely able to be used at present, and argued that semantics-mediated data integration may be an effective approach to integration, curation, and utilization. Finally, Natalio Krasnogor identified desktop-scale experimentation and instrumentation, which can displace lab-work as the critical path in engineering, as a key missing component that will likely heavily utilize specialized silicon products.

State-of-the-art and current challenges

While biological systems in general are amongst the canonical examples of massively complex systems, it is important to remember that engineering practices have been able to successfully manage other systems with massive complexity. Notable amongst those are two other canonical examples of massively complex systems: software engineering and the semiconductor engineering that underlies it. While there are many differences between these two areas, complexity management within them has come to share many key techniques and practices, such as formal verification, deep integration of testing into design workflows, and standards-driven integration across organizations. The complexity of individual cells is large but not obviously more so than the complexity of large-scale software engineering projects: consider, for example, the interaction networks for *E. coli* and the Linux kernel compared in Fig. 19 [Yan10]: while there are differences in emphasis and balance in the networks, the overall scale and meta-structure is not dissimilar.

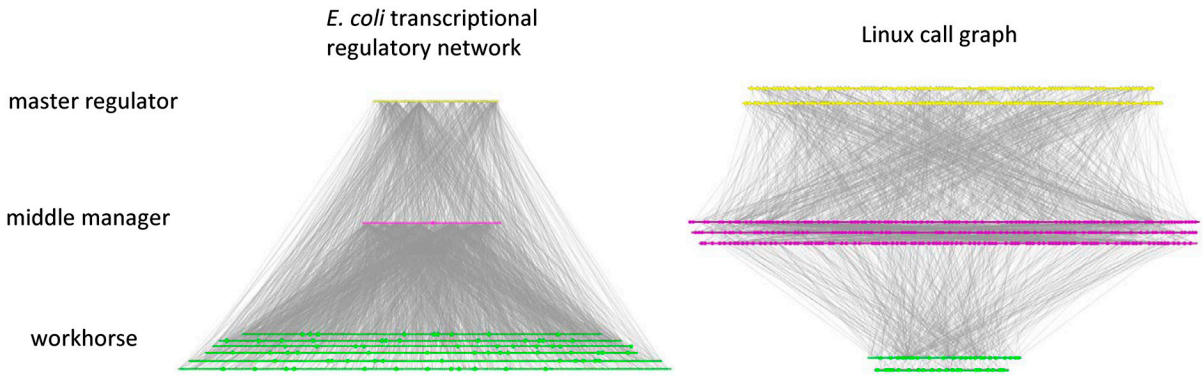


Figure 19: Transcriptional networks in E. coli are on a similar level of complexity to the Linux kernel call network (reproduced from [Yan10]).

Lower-level genetic design aspects of biological design have already drawn significantly from EDA, and will continue to do so [Myers16]. Many possible approaches to design automation have indeed been proposed, as discussed above in Session 1. Most, however, have drawn primarily on simple models of digital circuitry and Boolean logic. There is a rich world of work on asynchronous logic, stochastic computation, and analog computation that has not yet been well-explored and may offer much potential.

For the engineering of biological organisms to be effective and commercially realized, the engineering process must also support consideration of the full lifecycle, including deployment, maintenance, and disposal. It must be possible to evaluate how organisms are likely to behave under a wide variety of environmental conditions of operation. It must also be possible to predict and manage the possible autonomic evolution of the organisms over time, in response to the selective pressures of the engineered system incorporated within them and their operating context.

For performing such testing and validation of designs, the dominant approach to date has been to construct detailed biochemical models of cell behavior. Two key alternate approaches take a more abstract view of cell behavior: agent-based models abstract and approximate mechanisms, while formal verification performs model analysis at a more abstract mathematical level. Both of these approaches are already extremely well-developed in other domains.

Biochemical modeling has recently been able to model an entire simple cell [Karr12], *M. genitalium* (Fig. 20). This bacterium is one of the world's simplest living organisms, with only about one tenth as many genes as *E. coli* and far less than complex eukaryotes. The whole-cell model for *M. genitalium* has been able to effectively predict some cellular behaviors. Its implementation involves the integration of a large number of heterogeneous model components, enabled by SBML [Hucka03]. While successful, this project also illustrates a key limitation of biochemical modeling: it is difficult to scale both in complexity and execution speed. Making this model work has required hand-tuned representational choices, and the details of its quantitative relationship with actual cellular behavior are still being determined. Note that such modeling may become a significant target for co-development with specialized hardware to support faster simulations, though the degree of speed-up possible through such processes has not yet, to the best of our knowledge, been analyzed for whole-cell models.

One approach to the challenge of scalability is to simplify modeling by raising the level of abstraction, e.g., via agent-based models [Holcombe16, Bai14]. Agent-based models (and other similar model abstraction approaches) offer the potential to capture much more complex, often stateful, functions in

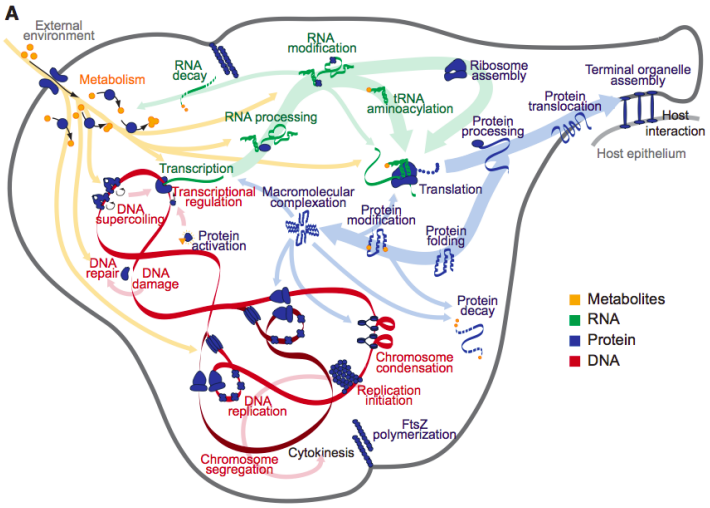


Figure 20: Modules of whole-cell model of *M. genitalium* (reproduced from [Karr12])

describing the elements of a system and their evolving interactions over time (Fig. 21). When such representations are properly tuned, this can allow a dramatic reduction in complexity with comparison to more fundamental physics- and chemistry-driven biochemical models. The complementary challenge is that configuring and validating such models is much more difficult, for the simple reason that, unlike physical or chemical models, there is no well-defined relationship of an agent to underlying physical processes. As such, to date agent-based models have typically been applied to particular specialized cases where their validity can be approximately established, rather than to more complex and holistic challenges such as whole-cell modeling.

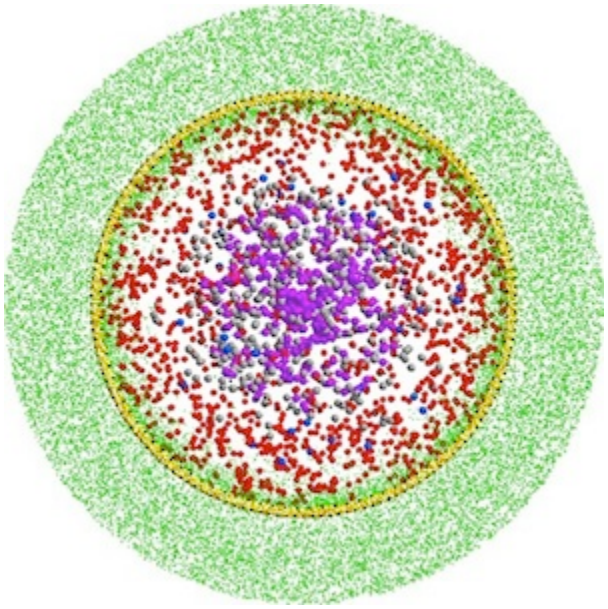


Figure 21: Agent-based model of E. coli interacting with oxygen (reproduced from [Bai14]

As an alternative to raising the level of abstraction in a model, one may instead raise the level of abstraction in how a model is applied and analyzed [Jha16]. Formal verification, such as the work presented in [Jha09], takes this approach, applying the arsenal of tools that have been constructed for verification and validation of software and hardware designs. Here, the underlying model is still likely to be physical or chemical in nature. Rather than analyzing the system by executing simulations using this model, however, these approaches analyze the model directly to determine its mathematical structure and whether it conforms to specified properties. One challenge in applying these approaches is that much of formal verification is predicated on assumptions of deterministic relations with a small number of states per variable, whereas biological systems often involve significant stochastic relations and large ranges of variable values (well-approximated in many cases by continuous ranges). A number of techniques likely to be relevant to such systems, however, have also already been developed. For example, stochastic model checking has already been applied to analysis of biological systems (e.g., [Calder06, Madsen14]) and fluid model checking being used for complex systems analysis in other fields [Bortolussi12] is likely to be of use in formal verification of biological systems as well.

No matter what technique is applied to analyze systems, it is also important to be able to scope what properties are to be analyzed and in what conditions. Here, synthetic biology appears likely to diverge strongly from systems biology in the type of questions that may be considered relevant to ask. Systems biology aims at a general understanding of the nature and behavior of biological systems, which is a very open-ended question. Any given synthetic biology engineering project, like any other engineering project, is likely to be focused on the satisfaction of a particular collection of specifications. Again, there is a well-developed set of techniques that might be drawn upon to build the testing and validation portions of biological engineering workflows, such as regression testing [Fisher04].

Just as in other engineering disciplines, testing will need to operate at different levels of abstraction. The more sophisticated the underlying models, the more precise the debugging that can be offered by such tests: black-box testing considers only the externally observable input/output relations of a system, grey-box testing uses some (generally abstract) knowledge of its contents, and white-box testing incorporates a full specification. Black-box testing is good for "sanity checks," but can never provide full validation of any but the simplest systems, and biological systems are not simple. White-box testing (e.g., unit testing) can potentially provide complex validation, but requires deep understanding of both system and specification. Grey-box testing provides an intermediate point with some of the advantages and disadvantages of both. At present, challenges of both modeling and test data availability limit most validation to black-box testing. As these are improved, it is likely that synthetic biology workflows will be able to progress to routinely incorporate grey-box testing and white-box testing as well [Krasnogor16-2].

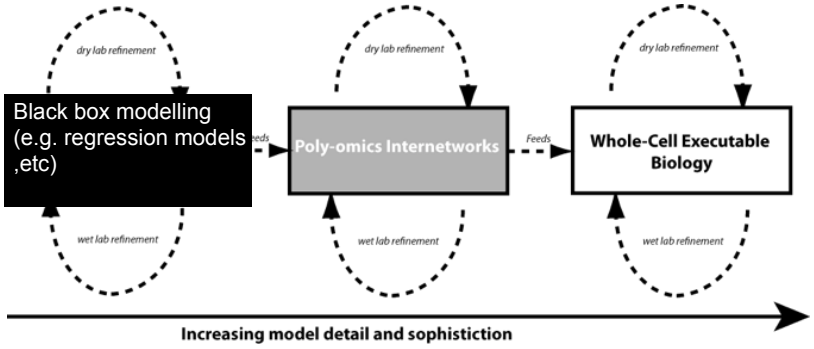


Figure 22: Better models and equipment can enable a shift from black-box to grey-box and ultimately white-box testing [Krasnogor16-2].

Data availability is a critical challenge for model validation [Wipat16]. Some of this challenge can be met by integration of existing data. The amount of biological assay data being gathered and stored is growing exponentially, providing a potentially vast amount of resources (Fig. 23). This data is scattered across many different databases and organizations, however, and its quality of curation and metadata is often highly variable. Because of such scale and heterogeneity, automation-assisted curation and integration of biological databases will be required in order to make effective use of these resources for biological engineering. Work is ongoing on such systems, based on existing and emerging standards [Misirli16, Roehner16].

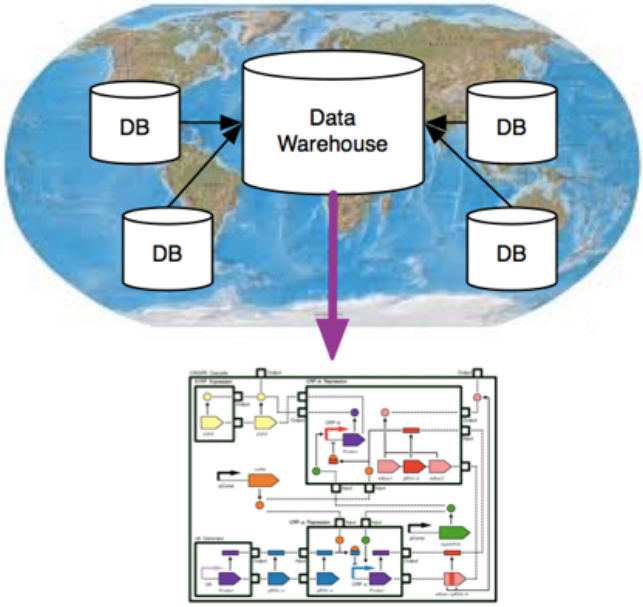


Figure 23: The rapidly growing collection of biological information scattered in various databases can be integrated with appropriate interchange standards for describing biological designs, evaluation contexts, and results [Wipat16].

Ultimately, however, the engineering of biological systems cannot depend on either opportunistic use of data gathered for other purposes or on slow and costly human-centric experimentation. Instead, biological engineering workflows will be supported by testing that is fast, low-cost, and tightly integrated into every aspect of the workflow. Agile development techniques that serve well in the software and hardware world, such as continuous integration and regression testing, will be needed to bring any complex system safely to deployment and to support maintenance and continued development of fielded biological systems. This will only be possible through vastly improved miniaturization and automation, such as was discussed in Session 2.

Session 4 Roundtable Discussion Summary

Discussions in this session centered around the following points:

- Software design automation and representation are already becoming quite important for design of biological systems, particularly with regards to standards for design representation and data exchange.
- Biological data is expanding rapidly in volume and scope but is suffering major problems in curation and integration. Better representations, likely assisted by machine learning and inference technologies, are needed in order to integrate this data and realize its potential.

- There are many interesting possible approaches to "higher-level" software design automation and representations of biological systems. The degree to which these can currently be developed, however, appears to be limited by current biological knowledge and bottlenecks in experimentation related to the current labor-intensive nature of laboratory work.

Forward Outlook

Analysis of the forward outlook for this topic is organized in several categories: (i) Major challenges and issues, (ii) Promising research topics, and (iii) Vision for the next 5 years, 10 years, and 15-20 years.

(i) Major challenges and issues

- Design needs to support consideration of the full lifecycle, including deployment, maintenance, and disposal.
- BDA needs to be able to effectively consider environmental interactions and the possible self-evolution of the system over time.
- Need for cheap "on the desktop" experimentation in the loop with design tools.
- Software tools to detect potential threats must be balanced with need for open interchange in a developing commercial ecosystem.
- Both business development and complexity management require development and adoption of standards and integration of disparate data sources.

(ii) Promising research directions/topics

- Standards and integration methods to enable BDA to effectively draw on existing and emerging biological databases.
- BDA tools outside of the simple digital paradigm.
- Application of "whole cell models" to predictive and precise engineering of novel organisms.
- Tools for monitoring, testing, and management of biological designs across multiple scales, from cells to tissues and organisms to ecosystem and society.
- Fast, cheap, low-scale build and test hardware (see Session 2) and its integration into biological engineering processes.
- Adaptation of agile development workflows for biological engineering.

(iii) Vision for the next 5 years, 10 years, and 15-20 years

5 years

- Comprehensive models, on the scale of a complete bacterium, integrated to support precision engineering.
- Integration of all major human-curated biological databases into an effective federated resource to support biological design.
- BDA tools based on asynchronous and stochastic computational abstractions.

10 years

- Comprehensive models, on the scale of a complete eukaryotic cell, integrated to support precision engineering.
- Automatic curation of biological databases

- Biological engineering informational costs dominate lab work costs.
- Biological engineering adopts agile software development practices such as test-driven development and continuous integration.

15-20 years

- Comprehensive models, on the scale of a complex many-tissue eukaryotic organism, integrated to support precision engineering.
- Agile development practices for biological engineering on same scale of complexity as agile software development.

References

[Agabi16] Agabi, Oshiorenoya, “Intelligence is Natural”, SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Bai14] Bai, Hao, Matthew D. Rolfe, Wenjing Jia, Simon Coakley, Robert K. Poole, Jeffrey Green, and Mike Holcombe. "Agent-based modeling of oxygen-responsive transcription factors in Escherichia coli." *PLOS Comput Biol* 10, no. 4 (2014): e1003595.

[Beal12] Beal, Jacob, Ron Weiss, Douglas Densmore, Aaron Adler, Evan Appleton, Jonathan Babb, Swapnil Bhatia et al. "An end-to-end workflow for engineering of biological networks from high-level specifications." *ACS Synthetic Biology* 1, no. 8 (2012): 317-331.

[Beal14] Beal, Jacob, Tyler E. Wagner, Tasuku Kitada, Odisse Azizgolshani, Jordan Moberg Parker, Douglas Densmore, and Ron Weiss. "Model-driven engineering of gene expression from RNA replicons." *ACS Synthetic Biology* 4, no. 1 (2014): 48-56.

[Beal15] Beal, J. "Signal-to-noise ratio measures efficacy of biological computing devices and circuits." Synthetic Biology engineering complexity and refactoring cell capabilities (2015): 82.

[Beal16] Beal, Jacob, Traci Haddock-Angelli, Markus Gershater, Kim de Mora, Meagan Lizarazo, Jim Hollenhorst, and Randy Rettberg. "Reproducibility of fluorescent expression from engineered biological constructs in E. coli." *PLOS ONE* 11, no. 3 (2016): e0150182.

[Bogdan16] Bogdan, Paul, “Probing the Computational Thinking in Biological Systems”, SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Bonnet13] Bonnet, Jerome, Peter Yin, Monica E. Ortiz, Pakpoom Subsoontorn, and Drew Endy. "Amplifying genetic logic gates." *Science* 340, no. 6132 (2013): 599-603.

[Bortolussi12] Bortolussi, Luca, and Jane Hillston. "Fluid model checking." In *International Conference on Concurrency Theory*, pp. 333-347. Springer Berlin Heidelberg, 2012.

[Brisk16] Brisk, Philip, “Recent Developments in Microfluidic Large Scale Integration”, SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Calder06] Calder, Muffy, Stephen Gilmore, and Jane Hillston. "Modelling the influence of RKIP on the ERK signalling pathway using the stochastic process algebra PEPA." In *Transactions on computational systems biology VII*, pp. 1-23. Springer Berlin Heidelberg, 2006.

[Carlson10] Carlson, Robert H. *Biology is technology*. Harvard University Press, 2010.

[Carr16-1] Carr, Peter, “EDA-BDA synergy”, SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Carr16-2] Carr, Swati Banerjee. "Reliable gene expression and assembly for synthetic biological devices in E. coli through customized promoter insulator elements and automated DNA assembly." PhD diss., 2016.

[Chandran09] Chandran, Deepak, Frank T. Bergmann, and Herbert M. Sauro. "TinkerCell: modular CAD tool for synthetic biology." *J. Biological Eng*. 3, no. 1 (2009): 19.

[Czar09] Czar, Michael J., Yizhi Cai, and Jean Peccoud. "Writing DNA with GenoCAD™." *Nucleic Acids Res*. 37, Suppl 2 (2009): W40-W47.

[Davidsohn14] Davidsohn, Noah, Jacob Beal, Samira Kiani, Aaron Adler, Fusun Yaman, Yinqing Li, Zhen Xie, and Ron Weiss. "Accurate predictions of genetic circuit behavior from part characterization and modular composition." *ACS Synthetic Biology* 4, no. 6 (2014): 673-681.

[Densmore16] Densmore, Douglas, "EDA - BDA Synergy", SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Fisher04] Fisher, Jasmin, David Harel, E. Jane Albert Hubbard, Nir Piterman, Michael J. Stern, and Naamah Swerdlin. "Combining state-based and scenario-based approaches in modeling biological systems." In *International Conference on Computational Methods in Systems Biology*, pp. 236-241. Springer Berlin Heidelberg, 2004.

[Gach16] Gach, Peter, "Microfluidic Devices for Automating Synthetic Biology", SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Gendrault11] Gendrault, Yves, Morgan Madec, Christophe Lallement, François Pecheux, and Jacques Haiech. "Synthetic biology methodology and model refinement based on microelectronic modeling tools and languages." *Biotechnology J.* 6, no. 7 (2011): 796-806.

[Hessel16] Hessel, Andrew, "Biology is the next design revolution", SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Holcombe16] Holcombe, Mike, "Software design automation for complex biological and electronic systems – *In Silico* Biosystems Design", SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Hoops06] Hoops, Stefan, Sven Sahle, Ralph Gauges, Christine Lee, Jürgen Pahle, Natalia Simus, Mudita Singhal, Liang Xu, Pedro Mendes, and Ursula Kummer. "COPASI—a complex pathway simulator." *Bioinformatics* 22, no. 24 (2006): 3067-3074.

[Hucka03] Hucka, Michael, Andrew Finney, Herbert M. Sauro, Hamid Bolouri, John C. Doyle, Hiroaki Kitano, Adam P. Arkin et al. "The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models." *Bioinformatics* 19, no. 4 (2003): 524-531.

[Jha09] Jha, Sumit K., Edmund M. Clarke, Christopher J. Langmead, Axel Legay, André Platzer, and Paolo Zuliani. "A bayesian approach to model checking biological systems." In *International Conference on Computational Methods in Systems Biology*, pp. 218-234. Springer Berlin Heidelberg, 2009.

[Jha16] Jha, Sumeet, "Algorithmic Synthesis in Biological Design Automation: Challenges & Some (Biased) Solution Templates", SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Karr12] Karr, Jonathan R., Jayodita C. Sanghvi, Derek N. Macklin, Miriam V. Gutschow, Jared M. Jacobs, Benjamin Bolival, Nacyra Assad-Garcia, John I. Glass, and Markus W. Covert. "A whole-cell computational model predicts phenotype from genotype." *Cell* 150, no. 2 (2012): 389-401.

[Kiani14] Kiani, Samira, Jacob Beal, Mohammad R. Ebrahimkhani, Jin Huh, Richard N. Hall, Zhen Xie, Yinqing Li, and Ron Weiss. "CRISPR transcriptional repression devices and layered circuits in mammalian cells." *Nature Methods* 11, no. 7 (2014): 723-726.

[Krasnogor16-1] Krasnogor, Natalio, "New Design Principles for Semiconductors systems inspired by Biological Systems", SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Krasnogor16-2] Krasnogor, Natalio, "Software design automation for complex biological and electronic systems", SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Li15] Li, Yinqing, Yun Jiang, He Chen, Weixi Liao, Zhihua Li, Ron Weiss, and Zhen Xie. "Modular construction of mammalian gene circuits using TALE transcriptional repressors." *Nature Chem. Biol*. 11, no. 3 (2015): 207-213.

[Lou12] Lou, Chunbo, Brynne Stanton, Ying-Ja Chen, Brian Munsky, and Christopher A. Voigt. "Ribozyme-based insulator parts buffer synthetic circuits from genetic context." *Nature Biotechnology* 30, no. 11 (2012): 1137-1142.

[Madec16] Madec, Morgan, "Adaptation of microelectronics design tools to synthetic biology", SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Madsen14] Madsen, Curtis, Zhen Zhang, Nicholas Roehner, Chris Winstead, and Chris Myers, "Stochastic Model Checking of Genetic Circuits, in ACM Journal on Emerging Technologies in Computing Systems", 11(3), December, 2014.

[Misirli16] Mısırlı, Göksel, Jennifer Hallinan, Matthew Pocock, Phillip Lord, James Alastair McLaughlin, Herbert Sauro, and Anil Wipat. "Data Integration and Mining for Synthetic Biology Design." *ACS Synthetic Biology* 5, no. 10 (2016): 1086-1097.

[Mutalik13] Mutalik, Vivek K., Joao C. Guimaraes, Guillaume Cambray, Colin Lam, Marc Juul Christoffersen, Quynh-Anh Mai, Andrew B. Tran et al. "Precise and reliable gene expression via standard transcription and translation initiation elements." *Nature Methods* 10, no. 4 (2013): 354-360.

[Myers16] Myers, Chris, "Software Design Automation for Complex Biological and Electronic Systems", SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Nielsen16] Nielsen, Alec AK, Bryan S. Der, Jonghyeon Shin, Prashant Vaidyanathan, Vanya Paralanov, Elizabeth A. Strychalski, David Ross, Douglas Densmore, and Christopher A. Voigt. "Genetic circuit design automation." *Science* 352, no. 6281 (2016): aac7341.

[Ostrov16] Ostrov, Nili, Matthieu Landon, Marc Guell, Gleb Kuznetsov, Jun Teramoto, Natalie Cervantes, Minerva Zhou et al. "Design, synthesis, and testing toward a 57-codon genome." *Science* 353, no. 6301 (2016): 819-822.

[Parsons16] Parsons, Gregory, "Computation with Living Cellular Systems", SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Perli16] Perli, Samuel, "Foundations and Emerging Paradigms for Computing in Living Cells", SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Richardson17] Richardson, Sarah M., Leslie A. Mitchell, Giovanni Stracquadanio, Kun Yang, Jessica S. Dymond, James E. DiCarlo, Dongwon Lee et al. "Design of a synthetic yeast genome." *Science* 355, no. 6329 (2017): 1040-1044.

[Roehner16] Roehner, Nicholas, Jacob Beal, Kevin Clancy, Bryan Bartley, Goksel Misirli, Raik Grunberg, Ernst Oberortner, Matthew Pocock, Michael Bissell, Curtis Madsen, Tramy Nguyen, Michael Zhang, Zhen Zhang, Zach Zundel, Douglas Densmore, John H. Gennari, Anil Wipat, Herbery M Sauro, and Chris J. Myers, "Sharing structure and function in biological design with SBOL 2.0", *ACS Synthetic Biology* 5 , no. 6 (2016): 498–506.

[Roquet16] Roquet, Nathaniel, Ava P. Soleimany, Alyssa C. Ferris, Scott Aaronson, and Timothy K. Lu. "Synthetic recombinase-based state machines in living cells." *Science* 353, no. 6297 (2016): aad8559.

[Shih15] Shih, Steve CC, Garima Goyal, Peter W. Kim, Nicolas Koutsoubelis, Jay D. Keasling, Paul D. Adams, Nathan J. Hillson, and Anup K. Singh. "A versatile microfluidic device for automating synthetic biology." *ACS Synthetic Biology* 4, no. 10 (2015): 1151-1164.

[Shuaib16] Shuaib, Aban, Adam Hartwell, Endre Kiss-Toth, and Mike Holcombe. "Multi-Compartmentalisation in the MAPK Signalling Pathway Contributes to the Emergence of Oscillatory Behaviour and to Ultrasensitivity." *PLOS ONE* 11, no. 5 (2016): e0156139.

[Stanton15] Stanton, Brynne C., Alec AK Nielsen, Alvin Tamsir, Kevin Clancy, Todd Peterson, and Christopher A. Voigt. "Genomic mining of prokaryotic repressors for orthogonal logic gates." *Nature Chem. Biol.* 10, no. 2 (2014): 99-105.

[Sukharev16] Suhkarev, Valeriy. "SemiSynBio: EDA ←→ BDA", SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Thorsen16] Thorsen, Todd. "Hybrid Microfluidic Design and Automation", SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Wei13] Wei, Guopeng, Paul Bogdan, and Radu Marculescu. "Efficient modeling and simulation of bacteria-based nanonetworks with BNSim." *IEEE Journal on Selected Areas in Communications* 31, no. 12 (2013): 868-878.

[Widdershoven16] Widdershoven, Frans. "Benefit from CMOS big time, but don't mess it up!", SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Wipat16] Wipat, Anil, “Challenges to software design automation for complex biological and electronic systems”, SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Yakovlev16] Yakovlev, Alex, “New Electronic Design principles inspired by bio systems.”, SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Yan10] Yan, Koon-Kiu, Gang Fang, Nitin Bhardwaj, Roger P. Alexander, and Mark Gerstein. "Comparing genomes to computer operating systems in terms of the topology and evolution of their regulatory control networks." *Proceedings of the National Academy of Sciences* 107, no. 20 (2010): 9186-9191.

[Zhang16] Zhang, Yong, “Learn from Neuron”, SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016

[Zhirnov16] Zhirnov, Victor, “EDA-BDA synergy”, SemiSynBio Workshop on EDA/BDA interaction Roadmap, Newcastle University, Newcastle upon Tyne, UK, August 19 & 20, 2016