frontiers in
BIOENGINEERING AND BIOTECHNOLOGY

Synthetic Biology

Bridging the Gap: A Roadmap to Breaking the Biological Design Barrier

Jacob Beal

Journal Name:	Frontiers in Bioengineering and Biotechnology
ISSN:	2296-4185
Article type:	Review Article
Received on:	24 Oct 2014
Accepted on:	21 Dec 2014
Provisional PDF published on:	21 Dec 2014
Frontiers website link:	www.frontiersin.org
Citation:	Beal J(2014) Bridging the Gap: A Roadmap to Breaking the Biological Design Barrier. *Front. Bioeng. Biotechnol.* 2:87. doi:10.3389/fbioe.2014.00087
Copyright statement:	© 2014 Beal. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution and reproduction in other forums is permitted, provided the original author(s) or licensor are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Frontiers in Bioengineering

Review Article
20 December 2014

Bridging the Gap: A Roadmap to Breaking the Biological Design Barrier

Jacob Beal [1,*]
[1] *Raytheon BBN Technologies, Cambridge, MA, USA*

Correspondence*:
Jacob Beal
Raytheon BBN Technologies, 10 Moulton Street, Cambridge, MA, USA,
jakebeal@bbn.com

ABSTRACT

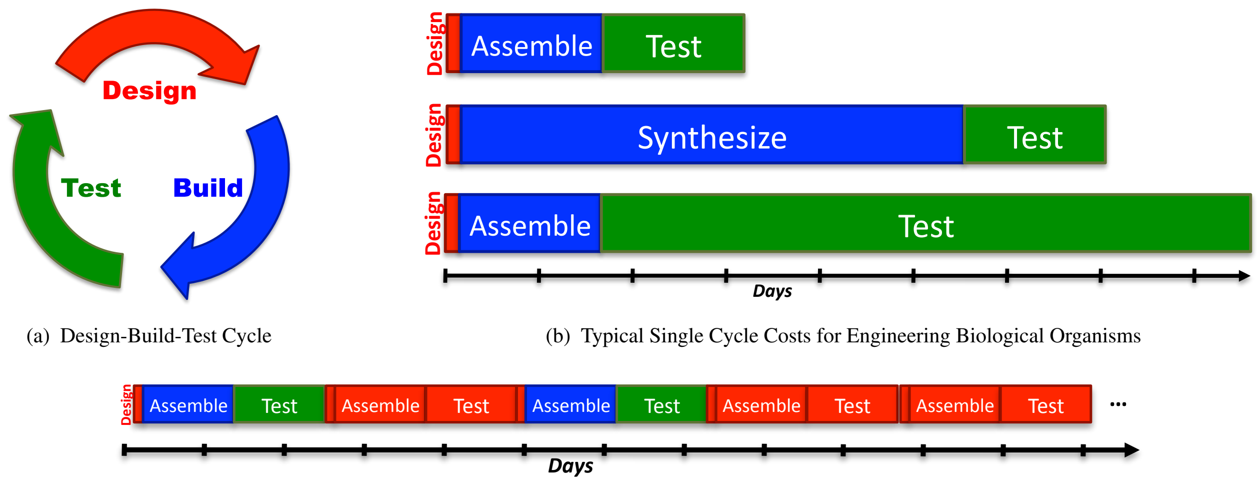
This paper presents an analysis of an emerging bottleneck in organism engineering, and paths by which it may be overcome. Recent years have seen the development of a profusion of synthetic biology tools, largely falling into two categories: high-level "design" tools aimed at mapping from organism specifications to nucleic acid sequences implementing those specifications, and low-level "build and test" tools aimed at faster, cheaper, and more reliable fabrication of those sequence and assays of their behavior in engineered biological organisms. Between the two families, however, there is a major gap: we still largely lack the predictive models and component characterization data required to effectively determine which of the many possible candidate sequences considered in the design phase are the most likely to produce useful results when built and tested. As low-level tools continue to mature, the bottleneck in biological systems engineering is shifting to be dominated by design, making this gap a critical barrier to progress. Considering how to address this gap, we find that widespread adoption of readily available analytic and assay methods is likely to lead to rapid improvement in available predictive models and component characterization models, as evidenced by a number of recent results. Such an enabling development is, in turn, likely to allow high-level tools to break the design barrier and support rapid development of transformative biological applications.

Keywords: Synthetic Biology, Organism Engineering, Design, Prediction, Automation, Metrology, Calibrated Flow Cytometry

1 INTRODUCTION

The ongoing revolution in synthetic biology is bringing about a fundamental transformation in our relationship with the world of living organisms. One of the drivers of this revolution is the exponential rate of improvement in our ability to sequence, synthesize, and deliver nucleic acid sequences (**Carlson**, 2011). Improvements in reading and writing nucleic acid sequences in turn enable increasingly rapid modification of an ever-broadening set of organisms using a growing toolkit of biological mechanisms. Another key driver is the ongoing adaptation of engineering concepts originating in computer science and electrical engineering (e.g., **Knight and Sussman** (1998); **Hasty et al.** (2002); **Knight** (2003); **Ferber** (2004); **Canton et al.** (2008)). In particular, methods for generating and exploiting abstraction and modularity have enabled a "component-based" approach to engineering biological organisms that greatly simplifies the isolation and dissemination of useful biological mechanisms.

Viewed through the lens of a "design-build-test" cycle of iterative engineering, the first set of advances addresses build and test, while the second set addresses design. In both areas, progress is rapid, and



(a) Design-Build-Test Cycle

(b) Typical Single Cycle Costs for Engineering Biological Organisms

(c) Notional Multi-Cycle Costs for Engineering Biological Organisms

Figure 1. When analyzing synthetic biology against a classic design-build-test cycle (a), analysis of a single cycle hides the cost of design, since the relative effort expended in a single cycle is typically quite low: (b) illustrates this with typical examples of time expended in a single cycle: a few minutes or hours of design, followed by construction of a design via BioBrick assembly (**Knight**, 2003) from existing sequence fragments or via an expedited order from a next-generation synthesis company, and then a 1-day test (e.g., a simple bacterial circuit) or a 1-week test (e.g., testing a memory circuit). A better measure, however, also takes into account the number of cycles required due to imprecision in design: (c) illustrates a sequence of design-build-test cycles in which any cycle that turns out not to be productive is accounted as a cost of problems in design.

advances are being encapsulated into tools that allow these improved methods to be widely applied. Between these two families of tools, however, there is a critical gap of growing importance: design, as it is currently typically practiced, is simply too imprecise. As has been widely and uncomfortably observed (e.g., **Kwok** (2010)), the behavior of engineered biological systems is frequently far different from predictions, typically due to some combination of unavailable information, inaccurate or incomplete models, and insufficiencies in available components. As a result, current practices for engineering biological systems typically require many iterations, and both time and cost increase rapidly with the complexity of the system to be engineered.

A number of recent results, however, indicate that these problems are becoming tractable to address. The goal of this manuscript is thus to analyze the problem landscape from a systems engineering perspective, producing a roadmap for breaking the biological design barrier and enabling the rapid and effective engineering of complex biological systems. Section 2 begins by analyzing the design-build-test cycle of biological engineering, developing an information-based metric for analyzing both the complexity of biological engineering problems and the efficacy of various engineering methodologies. Section 3 then applies this metric to analyze the relative potential impact from three complementary lines of tool development: high-throughput assays, improved device families, and predictive modeling and design. Narrowing the focus to predictive modeling, Section 4 examines the requirements necessary for an assay to effectively support predictive modeling, and illustrates how these can be satisfied through the example of a recently developed method for calibrated flow cytometry. Section 5 then connects assay capabilities back to the engineering of biological organisms by showing how calibrated flow cytometry has been used for intra-sample validation, to develop high-precision predictive models, and to support development of improved repressor devices. Finally, Section 6 summarizes and presents a roadmap for extending these capabilities to be readily applicable to a broad class of systems and organisms.

Symbol	Definition
T_x	Time to complete a particular portion x of a sequential process.
S	A specification setting requirements for a particular system to be engineered in an organism
$n_{cycles}(S)$	Number of Design-Build-Test cycles used to engineer a system satisfying S for a desired organism
C	Space of possible system configurations to be considered
G_S	Set of "goal" configurations satisfying specification S
w_i	Estimated probability at cycle i that configuration $c \in C$ is in G_S
$H(S,i)$	Information-based estimate of the remaining difficulty of engineering to specification S at cycle i
ΔH_i	Information gain, quantifying progress in the ith cycle.
$E[\Delta H]$	Expected information gain per cycle
$\hat{n}_{cycles}(S)$	Estimate of expected number of cycles required to engineer a given system
$\hat{T}_{total}(S)$	Estimate of expected time required to engineer a given system

Figure 2. Table of significant notation used for analysis of organism engineering bottlenecks.

2 QUANTIFYING BOTTLENECKS IN ENGINEERING BIOLOGICAL ORGANISMS

Engineering is often conceived of as an iterative "design-build-test" process (Figure 1(a)). Although more mature engineering disciplines typically develop more sophisticated workflows (e.g., continuous integration (**Duvall**, 2007) and Agile processes (**Larman**, 2004) in software design, design for test (**Crouch**, 1999) in electronics, waterfall processes (**INCOSE**, 2010) for complex electromechanical systems), the classic iterative model is a good starting point for analyzing current synthetic biology approaches to the engineering of biological systems. Taking this approach, we can apply Amdahl's law (explained in Section 2.1) to quantify process bottlenecks in the engineering of biological organisms. Here, the true cost of design becomes clear when it is considered as a multi-iteration search process, and can be estimated using the ratio of information required for a design to the information gained per test. This analysis can then be applied to assess the current ecosystem of design tools, identifying critical gaps and opportunities.

For the convenience of the reader, Figure 2 provides a table of notation used in this section.

2.1 QUANTIFYING BOTTLENECKS: AMDAHL'S LAW

In computer science, Amdahl's law (**Amdahl**, 1967) is frequently used for analyzing cost-benefit tradeoffs in optimizing the speed of complex processes. In essence, Amdahl's law is simple arithmetic: if a particular stage of a sequential process takes time T_{stage} to complete, and the remainder take a total time $T_{total} - T_{stage}$, then optimizing that stage to be k times faster gives an overall speed-up of:

$$k_{total} = \frac{T_{total}}{(T_{total} - T_{stage}) + \frac{T_{stage}}{k}} \tag{1}$$

In other words, the efficacy of improving one stage in a process (e.g., by increasing parallel throughput) is bounded by the fraction of time spent in the other stages.

For example if there are two stages, the first taking 2 days and the second taking 1 day, then a $k = 2$ speedup of the first stage, to one day, will improve the total speed by $\frac{3}{1+1} = 1.5$ times, while a $k = 2$ speedup of the second stage to half a day will only improve the total speed by $\frac{3}{2+0.5} = 1.2$ times.

Let us consider how this analysis applies to the design-build-test cycle (Figure 1(a)) for one of the most common process workflows in the practice of synthetic biology:

- **Design** proposes a set of nucleotide sequences that will be assayed with the aim of advancing toward some engineering goal.
- **Build** physically realizes the desired nucleotide sequences through some combination of protocols such as for synthesis, assembly, editing, and purification.
- **Test** assays the results of gene expression from these nucleotide sequences under various experimental conditions.

This general schema covers much of the practice of synthetic biology, from developing sensors to tuning chemical synthesis, from directed evolution to circuit engineering, from microbes to specialty mice.

A simplistic analysis could simply consider the time required for each of these stages in a cycle:

$$T_{cycle} = T_{design} + T_{build} + T_{test} \tag{2}$$

Applying Amdahl's law to this formula, we can determine where there is the most opportunity for improvement in a given engineering cycle. At present, this is typically dominated by building or testing, which frequently require days to weeks, depending on the particular constructs and organisms involved, as illustrated in Figure 1(b). From the single-cycle perspective, then, design would appear to be only a niche concern, relevant only for subareas with particularly high computational requirements, such as rational design of proteins.

Since there may be many cycles, however, a better approximation of the time required for engineering a biological organism to meet some particular specification S is:

$$T_{total}(S) = n_{cycles}(S) \cdot T_{cycle} \tag{3}$$

The number of cycles, in turn, is affected by the *quality* of choices made during each design phase. For example, design choices that lead to dead ends or simply turn out to fail may result in unproductive cycles that may reasonably have their costs assigned to design, as illustrated in Figure 1(c).

Since every engineering project is likely to face different challenges, how can we analyze the effect of design methods on the number of cycles? At a fine grain, of course, we cannot predict which cycles will be unproductive—otherwise, they would not happen in the first place. We can, however, quantify the efficacy of any engineering method by viewing the sequence of design-built-test cycles as an incremental search through the space of possible designs. Viewed in this way, the *expected* performance of an engineering method can be analyzed using various well-established methods (**Russell and Norvig**, 2003) from artificial intelligence and information theory.

2.2 INFORMATION-BASED ESTIMATION OF ENGINEERING CYCLES

Let us consider the engineering process as a search through the space of possible designs. This design space C consists of the set of all possible system configurations within the scope of consideration. For any given design specification S, the subset $G_S \subseteq C$ is the set of "goal" configurations that sufficiently satisfy the specification (we will assume these are simple to recognize when tested). The search process is thus an attempt to either identify at least one member of G_S or to determine that the set is empty.

For example, consider engineering of a metabolic pathway that expresses five enzymes: if each enzyme's expression is driven by a constitutive promoter and 5'UTR chosen from rationally engineered libraries of 10 of each type (e.g., via **Salis et al.** (2009)), and these functional units are joined to form a single plasmid, then there are $10^5 \cdot 10^5 \cdot 5! \cdot 2^5 = 3.8 \times 10^{13}$ possible configurations (10^5 for five independent choices from a set of 10 promoters, times 10^5 for five choices from 10 5'UTRs, times 5-factorial possible orderings, times 2^5 for five independent choices of plus or minus strand). For another example, consider engineering a circuit of 7 repressors drawn from the orthogonal library of 20 in (**Stanton et al.**, 2014). Selecting the repressors and organizing their functional units on a plasmid gives $\binom{20}{7} \cdot 7! \cdot 2^7 = 5.0 \times$

10^{10} possible configurations (there are $\binom{20}{7}$ possible combinations of library repressors, times 7-factorial possible orderings, times 2^7 for seven independent choices of plus or minus strand).

Prior knowledge about the likelihood of configurations being in G_S can be modeled by a normalized weight function $w_0 : C \to [0, 1]$, such that configurations known to not be in G_S map to 0 and those most likely to be in G_S map to 1. After each cycle, this function is updated to a new w_i based on the information learned from that cycle's tests. Any cyclic engineering process may then be modeled by the following meta-algorithm:

1. Select a set of candidate configurations $c \subseteq C$, on the basis of w_i.
2. Build and test all members of c.
3. If some $c \in G_S$, then **SUCCEED**
4. Incorporate knowledge gained from the tests to generate w_{i+1}.
5. If w_{i+1} is uniformly zero, then **FAIL**, since G_S has been demonstrated to be empty. Otherwise, return to step 1.

Using information theory, we can quantify how hard it is to find goal configurations by considering the selection of candidates at random.[1] By the standard definition of entropy, the number of bits $H(S, i)$ for the ith cycle of design toward some specification S is thus:

$$H(S, i) = \log_2\left(\frac{\sum\limits_{c \in C} w_i(c)}{\sum\limits_{c \in G_S} w_i(c)}\right) \tag{4}$$

where $w_i(c)$ is the estimated likelihood of configuration c belonging to G_S given the information available at cycle i. This is thus an *information-based* measure of the progress of engineering a system over time.

Note that better information about the likelihoods of configurations belonging to c reduces the number of bits, until in the limit w_i puts a non-zero weight only on members of G_S. At this point the number of bits is zero and success is certain. Complementarily, lack of information and sparse goals increase the number of bits toward an upper limit of $\log_2 |C|$. Thus the metabolic pathway example above is an engineering problem of up to 45.1 bits and the circuit example up to 35.5 bits.

The efficacy of an engineering method may then be evaluated in terms of the number of bits of information obtained per cycle, formally:

$$\Delta H_i = H(S, i) - H(S, i + 1) \tag{5}$$

In general, the number of bits remaining should decrease with each additional assay, giving a positive ΔH_i, and the greater the decrease in entropy, the better the efficacy of the method.

To illustrate these notions of information gain, consider the metabolic pathway example, beginning with no information about the appropriate expression levels. Some examples of information gain:

- Determining that enzyme #2's 5'UTR should be in the upper half of the expression range gains 1 bit of information.
- Determining which promoter should be used for enzyme #2 gains 3.32 bits of information.

[1] The assumption of random exploration may at first seem strange to those unfamiliar with information theory, since any engineering effort is of course guided by knowledge, experience, and educated guesses. In this information-based formulation of the engineering problems, however, all such guidance is encoded by the weight function. Whatever choice remains is arbitrary and cannot in general outperform random choice.

- Determining that enzymes #3 and #4 should be expressed with matching promoter/5'UTR combinations gains 6.64 bits of information.
- Using insulators that eliminate the effect of ordering and strand choice gains 11.91 bits of information.
- High-throughput screening of 10^{10} arbitrarily chosen combinations gains only 0.0004 bits of information.

Notice that in these examples the model-driven information gains are much greater than the gains from brute force screening—even with a rather large throughput. The relative balance of model-driven and exploration-driven approaches depends on the scale of the problems. For example, if the pathway contained only three enzymes rather than five, it would only be a 25.5-bit configuration space and could be screened completely using less than 10^8 combinations. Notice also, that some of these information gains are "one shot" while others are not. For example, using insulators is a single choice, and cannot narrow the design space further. A method that can incrementally refine expression level choices, however, might be applied iteratively to complete the entire design, providing a consistent expected information gain $E[\Delta H]$ per cycle.

A conservative estimate of the expected number of cycles $\hat{n}_{cycles}(S)$ required to engineer specification S by a particular engineering method with an expected information gain of $E[\Delta H]$ per cycle may thus be computed by assuming there is only a single possible solution in G_S. Under this assumption, the estimated number of cycles is:

$$\hat{n}_{cycles}(S) = \frac{H(S,0)}{E[\Delta H]} \tag{6}$$

Returning to the original equation, we may thus estimate the expected time to required to engineer a biological system to satisfy a given specification S as:

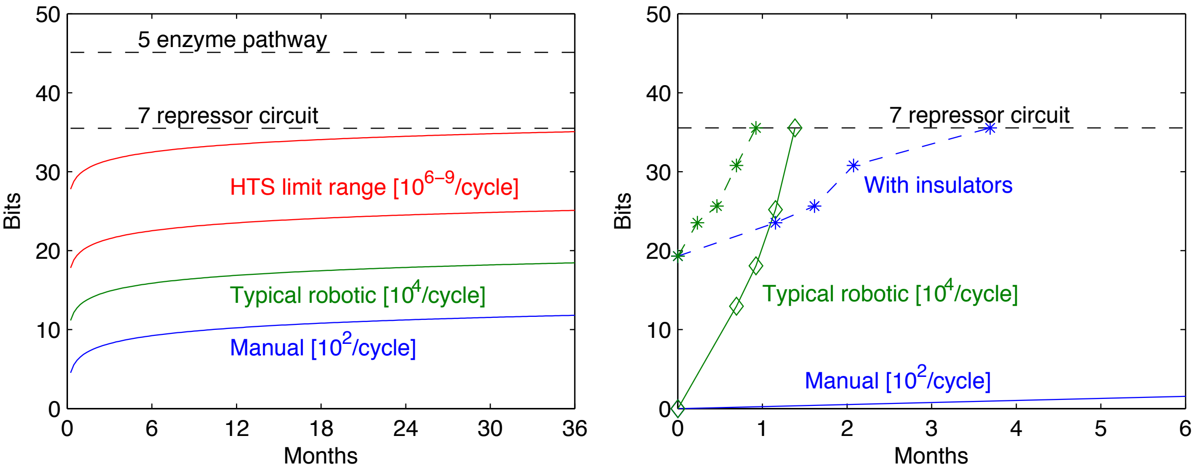
$$\hat{T}_{total}(S) = \frac{H(S,0)}{E[\Delta H]} \cdot T_{cycle} \tag{7}$$

3 POTENTIAL IMPACT OF IMPROVED ENGINEERING TOOLS

Let us now bring this analysis back to the original question: what are the bottlenecks in engineering biological organisms, and the key points for investigation to improve the situation? Having cast the problem of engineering biological organisms in information-based terms, we can see that there are only three terms in the equation for the estimated time for engineering. Each term then implies a particular strategy for improving speed:

1. decreasing the amortized time per configuration assay by decreasing amortized T_{cycle} (e.g., by decreasing the time per cycle or by running more cycles in parallel),
2. decreasing the effective bits $H(S,0)$ required by vastly enriching the number of acceptable "goal" configurations, and
3. increasing the number of bits $E[\Delta H]$ of design-constraining information expected to be gained per configuration assay.

The first strategy focuses on the "low level" tools aimed primarily at the build and test aspects of the engineering cycle. Capabilities in this area are increasing rapidly, but there are sharp limits in what can be enabled by this strategy alone. The second and third strategies relate more to "high level" design tools for mapping from a specification to a candidate configuration in C. Here there is a critical gap stemming



(a) High-Throughput Screening

(b) Design-Assisted Engineering

Figure 3. Decreased time per assay has sharply limited benefits, as illustrated by comparison of the bit size of example moderate-complexity circuits to rates of configuration space exploration (a). Solid lines show the number of bits of configuration space that can be assayed in a sequence of one-week cycles with various methods (starting with a single parallel assay at week one), while dashed lines show the complexity of the example circuits in Section 2.2. Improving models and components can dramatically reduce the required number of assays: (b) illustrates how a model-driven design process for the seven-repressor circuit might progress incrementally by breaking the system into three sequentially engineered subsystems (solid lines with diamonds marking sequence steps; manual is blue, robotic is green), and how that might be further improved with insulators that eliminate the effect of ordering and strand choice (dashed lines with stars marking sequence steps; manual is bue, robotic is green). For (b), progress toward completion is shown by graphing $H(S, 0) - H(S, i)$ for each circuit. Note that for the lowest manual line, no diamond appears because the first step is not completed for more than six months.

from the difficulty of predicting the behavior of a configuration, which we analyze along with emerging opportunities for bridging this gap.

3.1 DECREASING TIME PER ASSAY: HIGH-THROUGHPUT SCREENING

The time required for a cycle of testing is strongly limited by the underlying physical processes involved in build and test. Even if fabrication time might be greatly reduced, the time for an *in vivo* assay is an immutable bottleneck set by the inherent dynamics of the organism being assayed. This can potentially be mitigated through *in vitro* assays, assuming there are models and design tools that can mitigate the effect of differences between *in vitro* and *in vivo* environments. However, even the fastest *in vitro* assays (e.g., **Sun et al.** (2013); **Carlson et al.** (2012)), which operate on a time scale of only hours, are physically limited by the dynamics of the systems to be assayed. Thus, serial improvements to throughput are likely limited to around one order of magnitude increase in throughput relative to typical cell culture assays.

For the complementary approach, decreasing the amortized time per configuration assay through massive parallelization of high-throughput screening, typical practice is much farther from physical limits, giving the opportunity for much more improvement in throughput. High-throughput parallel screening is already a subject of much investigation, a recent review of which may be found in (**Dietrich et al.**, 2010). Significant research and commercial development is being invested in multiple tools designed to increase both capabilities and accessibility, including in robotics (e.g., **Linshiz et al.** (2012); **Hillson et al.** (2012); **Vasilev et al.** (2011)), microfluidics (e.g., **Kong et al.** (2007); **Gulati et al.** (2009)), evolutionary methods (e.g., **Esvelt et al.** (2011); **Lynch and Gill** (2012); **Cobb et al.** (2012)) and languages for low-level specification and data exchange (e.g., **Bilitchenko et al.** (2011); **Galdzicki et al.** (2012); **Myers** (2013)). This portion of the tool ecosystem is thus quite healthy and rapidly evolving.

Increasing throughput, however, has sharp limits in efficacy, because the configuration space grows exponentially with the number of bits, which in turn typically scales linearly with the complexity of the system being engineered. As a result, "brute force" approaches through high-throughput screening can readily solve problems up to a certain number of bits, but are effectively useless when addressing problems only a little bit larger. Figure 3(a) illustrates this scaling problem by comparing the circuit and metabolic pathway examples from the previous section with the size of configuration space that can be explored with a one week build/test cycle at various rates of high-throughput screening. When samples are prepared manually, the rates that can be effectively sustained for a single laboratory worker are on the order of 10^2 configurations per cycle (e.g., a few replicates in 96-well plates). A well-pipelined fluid handling robotics cell can prepare such assays continuously, raising the rate to around 10^4 configurations per cycle. Other techniques can potentially raise the rate by orders of magnitude, but there are limits due to various pragmatic barriers: (**Dietrich et al.**, 2010) calculated effective parallelism limits of high throughput screening to be around 10^6 to 10^9 configurations per screening assay. As can be seen, even at high rates of throughput, only moderately complex configuration spaces can be explored in a reasonable period of time.

High-throughput screening is still a valuable component of the toolkit for engineering biological organisms. As the capacity and accessibility of high-throughput screening continue to increase, brute force screening is likely to be sufficient for realizing a large number of "low hanging fruit" applications, particularly certain classes of medical, sensor, and chemical synthesis applications where the engineered organism only needs to operate for a relatively short period of time in a tightly controlled and isolated environment.

When contemplating longer lived systems in less controlled environments, however, it is reasonable to expect that there will generally be a need for more complex mechanisms that can ensure safety, stability, and effective operation under a range of conditions. Evolutionary methods are often proposed as a means of obtaining continuous incremental improvement toward such more complex systems (e.g., **Lynch and Gill** (2012); **Cobb et al.** (2012)). Great progress has been made with these methods, and they have proven highly effective for solving problems with simple specifications in permissive spaces (e.g., **Chudakov et al.** (2010); **Brustad and Arnold** (2011)). Computer science researchers, however, have identified a key set of open problems that must be addressed in order for evolutionary methods to be effective for complex engineered systems of any sort, biological or otherwise (**Forrest and Mitchell**, 1993; **ONeill et al.**, 2010). It is not yet clear whether it is even possible to solve these problems for synthetic biology, and unless they can be solved, effective engineering of even moderately complicated synthetic biology systems will necessarily require methods that include a more model-driven approach to design.

3.2 DECREASING REQUIRED ASSAYS: IMPROVING COMPONENTS AND MODELS

The other half of Equation 7, estimating number of cycles, addresses how large the effective configuration space is and how effectively an engineer can apply assays in searching for a goal configuration. Here, there is a major and and well recognized gap in the current tool ecosystem: given the current set of available parts and models, it has not generally been possible to accurately predict the behavior of multi-element designs except in certain special cases (**Kwok**, 2010; **Lux et al.**, 2012).

At higher levels of abstraction, mapping from behavior specifications for cell aggregates or individual cells to specifications of candidate regulatory networks intended to implement those specifications, there are a number of candidate tools and approaches. These encompass a wide variety of models, addressing computation and control, metabolic synthesis, and even the development of structure and patterns (e.g., **Pharkya et al.** (2004); **Beal and Bachrach** (2008); **Pedersen and Phillips** (2009); **Beal et al.** (2011); **Moriya et al.** (2010); **Marchisio and Stelling** (2011); **Huynh et al.** (2013); **Yousofshahi et al.** (2011)). Despite some notable current gaps,[2] this portion of the design space appears to be generally susceptible to a variety methods from computer science, electronic design automation, and signal processing.

[2] For example, most tools for designing computational circuits have focused on digital logic, rather than analog or hybrid systems.

Because of the difficulty in predicting the behavior of multi-element designs, however, no high-level design tool is currently capable of supporting an effective search of a large configuration space. Instead, at present, these tools generally either stop at the design of an "abstract" circuit that can be realized into many different configurations (e.g., **Beal et al.** (2011)), require collections of devices and/or data about devices that are not currently available (e.g., **Yaman et al.** (2012); **Pedersen and Phillips** (2009); **Huynh et al.** (2013); **Rodrigo et al.** (2011)), or can generate of large numbers of candidate configurations with no reliable means of distinguishing which (if any) are likely to actually meet the specification (e.g., **Bilitchenko et al.** (2011); **Czar et al.** (2009)). In all cases, however, the fundamental problem is a lack of precision in the available models for crossing the gap between a sequence specification and a prediction of the expression that will result from this sequence in context.

There are two basic approaches to addressing this problem, corresponding to the numerator and denominator of the estimated number of cycles. The first approach is to decrease the effective complexity of the configuration space $H(S, 0)$ by some combination of:

1. increasing the signal-to-noise characteristics of intended component interactions, and
2. decreasing the effect of unintended interactions between components and their environment.

Improvements of either type decrease the degree of coupling between choices in a configuration, i.e., the likelihood of incompatibility between choices: the lower the likelihood of two independent design choices being contained within G_S, the higher the degree of coupling between choices, because any given design choice must more carefully take into account the other choices that have been made. Coupling and effective complexity have a well-established relationship in both complexity theory (**Hogg et al.**, 1996; **Kanefsky and Taylor**, 1991) and statistical physics (**Krzakala and Kurchan**, 2007; **Zdeborová**, 2008; **DallAsta et al.**, 2008): as degree of coupling decreases, the structure of the configuration space undergoes a dramatic phase transition, such that it becomes easy to either find a goal configuration or determine that none exists. Put more intuitively: it is much easier to engineer with components that are less delicate.

In synthetic biology, a number of different ongoing efforts are aimed at improving signal-to-noise and at decreasing unintended interactions. Methods for improving signal-to-noise are currently largely focused on improving the number of available orthogonal high-amplification regulatory mechanisms. A number of approaches are being pursued, including recombinases (e.g., **Bonnet et al.** (2013)), homolog mining (e.g., **Stanton et al.** (2014)), and high-performance synthetic repressors (e.g., **Kiani et al.** (2014)). At the same time, methods for decreasing unintended interactions are being investigated across a large range of targets, from decreasing promoter/5'UTR interaction (e.g. **Lou et al.** (2012); **Mutalik et al.** (2013)), to making interactions between functional units more predictable by cotransfection (**Beal et al.**, 2014, 2012), to construction of entirely orthogonal systems of transcriptional machinery (**Neumann et al.**, 2010; **Schmidt**, 2010). It is unclear at present, however, how quickly these efforts can progress, how well their goals can be achieved, and in which classes of organisms.

The complementary approach to reducing $H(S, 0)$ is increasing $E[\Delta H]$, so that the search for functional configurations can be better guided by improved models of components and their intended and unintended interactions. More precise predictive models can improve the rate of information gain in a number of ways, notably including:

- Entire subspaces of non-functional configurations can be eliminated from consideration without any assay.
- An assay of one configuration can provide information (adjustments to w_i) about a large family of related configurations.
- Complicated systems can be decomposed hierarchically or thematically into subsystems whose details can be designed independently.

Figure 3(b) illustrates an example of how a model-driven engineering process might exploit such techniques, using the example of a seven-repressor circuit from Section 2.2. For this example, let us assume the same assay rates as for the high-throughput screening comparison in Figure 3(a), but instead of a brute-force search of the space, the circuit is broken into three modular subsystems and each of these engineered sequentially. First, all possible implementations of a subsystem of three repressors are assayed, followed in turn by two more two-repressor subsystems, covering the circuit. Note that the information gain for each stage is not uniform, because of how the combinatorics of possible remaining options differ. The repressors are then assayed for signal levels and orthogonality, and this information used to pick the best three compatible candidates for each subsystem: all combinations of these candidates are constructed, and the best version accepted, assuming is it sufficiently functional. Such a process may not identify the optimal system, but instead makes incremental progress towards identifying a “good enough” system, as long as the models are predictive enough and component coupling low enough to enable the subsystem assays to effectively constrain the candidates for the final design. The example design would still be too complex to search effectively with manual assays (an expected time of nearly five years at the specified rate), but can be readily tackled with fluid-handling robotics. Additional improvements to the design space might further improve the effective bits per assay: for example, with insulators that eliminate the effect of ordering and strand choice, even manual assay preparation is a viable strategy.

Construction of predictive models to enable such modular approaches to engineering has been a major goal of synthetic biology since its inception (e.g., **Weiss** (2001); **Knight and Sussman** (1998); **Elowitz and Leibler** (2000); **Gardner et al.** (2000)). Significant progress has been made in predictive engineering of behaviors of individual circuit components (e.g., **Salis et al.** (2009); **Lou et al.** (2012); **Liu et al.** (2012); **Borujeni et al.** (2013); **Kosuri et al.** (2013)). Successful prediction results for the interaction of multiple components, however, has been rare and generally applicable only to special cases (e.g., **Rosenfeld et al.** (2007); **Stricker et al.** (2008); **Tabor et al.** (2009); **Ellis et al.** (2009))

In the next section, we will argue that a major barrier to progress in predictive modeling has been the unavailability of sufficient assay methods. Recent improvements in assay methods, however, have enabled previously unattainable precision in quantification. Improved precision then enables better predictive models, supporting new and more effective approaches to both the engineering of multi-component circuits (**Davidsohn et al.**, 2014; **Beal et al.**, 2014) and to the engineering of individual components (**Kiani et al.**, 2014). All of this amounts to a significant increase in the amount of information that can be gained per assay, entirely complementary to high-throughput screening and component improvement, and offers the opportunity for rapid advancement in the speed of biological organism engineering.

4 MEASUREMENT ASSAYS TO SUPPORT MODEL-DRIVEN ENGINEERING

Let us now focus on the foundation for model-driven engineering: sufficiently powerful measurement assays. Effective quantitative modeling is impossible without being able to obtain accurate and precise measurements of the phenomena to be modeled. This section thus first analyzes what is required for an assay aimed to support model-driven engineering, then presents in detail an example of a recently developed method, calibrated flow cytometry, that satisfies these requirements.

4.1 ASSAY REQUIREMENTS FOR EFFECTIVE MODELING

Any effective synthetic biology program of quantitative modeling requires assays with the following capability: *absolute unit measurements from large numbers of single cells.* To see why, let us break this statement up and consider it one point at a time.

Absolute Unit Measurements Much of the prior work in both systems and synthetic biology reports results in relative or arbitrary units—in other words, values that are not tied to any SI unit. This is an unusual

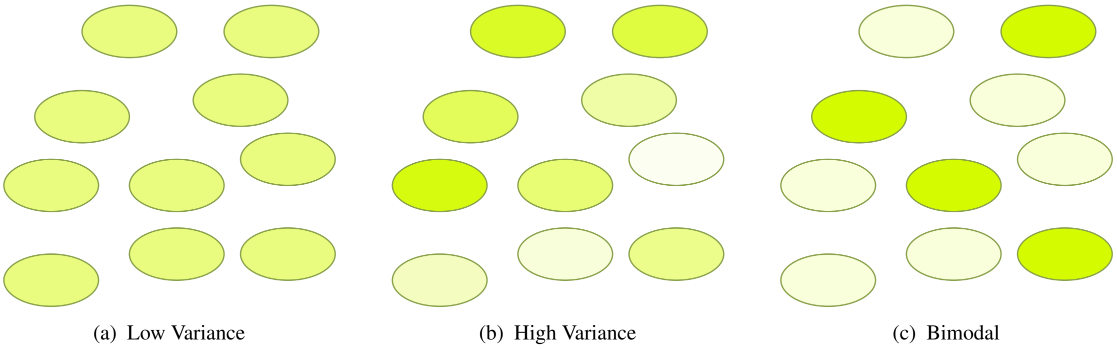


Figure 4. Assays that measure population means or totals cannot distinguish between even radically different distributions of expression. For example, the tight (a) broad (b) and bimodal (c) distributions illustrated above all have the same mean and total fluorescence.

practice for a scientific field, but so widespread that is goes virtually unremarked upon. When relative units are used, the focus of scientific reproducibility is not the individual measurements, but their relationship, e.g., the fold-repression exhibited by a transcription factor.

Relative units, however, cannot be combined across different experiments. This means that models of individual components or interaction phenomena cannot in general be combined to predict the behavior of new configurations. For quantitative models to be portable across experiments, systems, and laboratories, they must therefore be based on measurements tied to some absolute standard, preferably in SI units.[3]

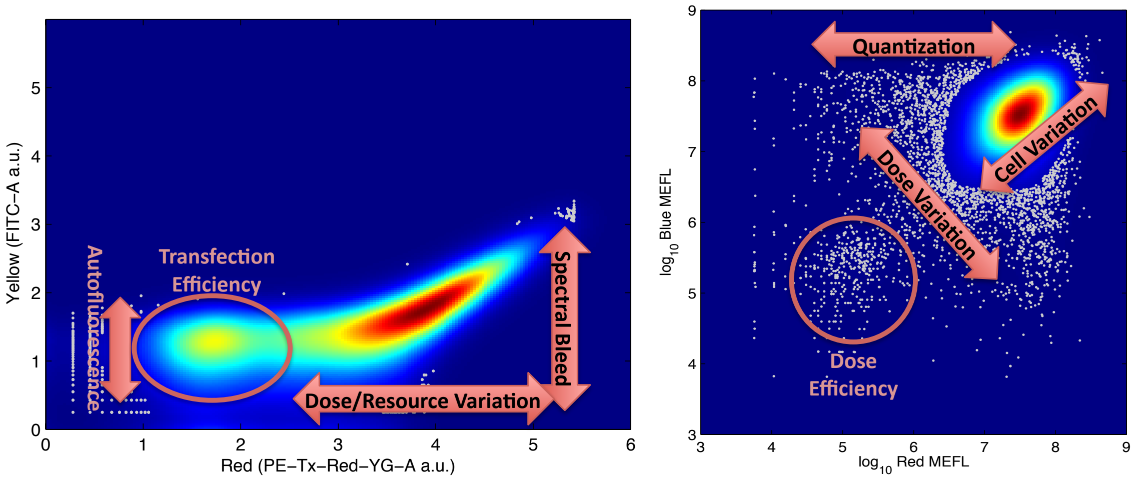
Single Cells In many synthetic biology systems, there is significant variation in the behavior of individual cells. Many assays, however, obtain only a cumulative or mean measurement across an entire population of cells. As a result, such population-level assays cannot distinguish between radically different distributions of values. For example, Figure 4 illustrates three very different distributions of fluorescence: a tight homogeneous distribution, a highly variable unimodal distribution, and a strongly bimodal distribution, all of which have the same mean and total fluorescence over the population. Since high cell-to-cell variation is so common, and has been shown to be important in understanding many systems (e.g., **Rosenfeld et al.** (2005); **Beal et al.** (2014)), effective quantitative modeling requires assays that can obtain their absolute measurements from individual cells.

Large Numbers of Cells Finally, not only is it important to take measurements of individual cells, but to obtain them from *large* numbers of individual cells. The reason is that there are often multiple different phenomena driving different modes of variation in the behavior of a population of cells. Some of the key classes of phenomena driving variation include:

- Inherent process stochasticity: e.g., transcription, translation, replication
- Cell-to-cell differences: e.g., size, cycle state, health, mutations, location
- Protocol stochasticity: e.g., transfection variation, insertion site
- Protocol execution issues: e.g., reagent variation, contamination, instrument drift

In modeling and engineering a system, each of these classes must be handled differently. For example, inherent stochasticity has largely uncorrelated effects on individual genetic components, while cell-to-cell differences have highly correlated effects on all of the components within an individual cell. Likewise, protocol stochasticity can often produce distributions of individual cell behaviors predictably controlled

[3] The RFU method (**Kelly et al.**, 2009) attempts to avoid this necessity by comparing with a standard constitutive promoter; the behavior of that promoter, however, may change radically from context to context.



(a) Constitutive transfection

(b) Dual replicon transfection

Figure 5. Examples of flow cytometry data showing complex population variation driven by multiple phenomena, with labels on key portions of the distribution used for estimating model parameters: (a) transient transfection of a single constitutively expressed fluorescent protein, from **Adler et al.** (2014), and (b) cotransfection of two replicons with constitutively expressed fluorescent proteins, from **Beal et al.** (2014). Both graphs indicate distribution density with color, with dark red indicating the maximum density, and outlier data (those in areas with less than 5% of maximum density) indicated by grey dots.

by variables in the protocol, while protocol execution issues are more generally unpredictable and must be detected and appropriately compensated for.

It is for this reason that large numbers of individual cell measurements are needed, in order to be able to accurately distinguish and resolve multiple modes of variation. Figure 5 shows illustrative examples of complex distributions of cell behaviors, labelled to indicate aspects of the distribution that can be used to quantify various significant mechanisms for understanding system behavior. For example, Figure 5(a), from (**Adler et al.**, 2014) shows transient transfection of a plasmid constitutively expressing red mKate fluorescent protein into HEK293 mammalian cells. Fluorescence is measured on two channels of a flow cytometer, one configured to measure mKate, the other to measure the yellow EFYP fluorescent protein. The distribution is strongly bimodal, with non-transfected cells expressing little red and transfected cells strongly expressing red: the relative number of non-transfected cells is proportional to transfection efficiency, while the range of expression in the transfected population indicates the range of variation in dose and resources from cell to cell. At the same time, the second channel can be used to quantify both autofluorescence (from the non-transfected cells) and the degree of spectral overlap between yellow and red channels of the instrument in its current state (from the location of the inflection point in the transfected population). Similar mechanisms are at work in Figure 5(b), taken from the data of (**Beal et al.**, 2014), showing a cotransfection of two Sindbis RNA replicons into BHK-21 mammalian cells, one constitutively expressing mKate, the other EBFP2. The relative size of the lower-left population of cells expressing minimal fluorescence is proportional to the transfection efficiency, and cell-to-cell variation in resources is proportional to the variance along the diagonal axis. In addition, the number of cells and variance of the off-axis component of the distribution indicate the size and distribution of the initial dose, since all doses are quantized (i.e., it is not possible to transfect a fractional replicon) and smaller initial doses have higher variance.

The exact number of observations required to quantify mechanism models from distributions such as these depends on the structure of the distributions involved. In general, however, more samples are required to obtain the same level of accuracy for more complex distributions or distributions with less clearly separated components. To give a sense of scale, the experiments reviewed in this section and the next range from around 30,000 to 1,000,000 samples per condition assayed, depending on the particular goals and requirements of the assay.

From these arguments and examples, we can see that an assay that can obtain absolute measurements from a large number of single cells has the potential to provide a great deal of insight into the behavior of a biological system. Moreover, it is likely to be difficult to make accurate predictive models of cell behavior without being able to use such a capability to separate and quantify the different modes of variation affecting cell behavior.

4.2 CALIBRATED FLOW CYTOMETRY

Until recently, there has been no readily accessible assay for gene expression that could satisfy the requirement for absolute measurements of large numbers (on the order of 10^5) of single cells. This has changed, however, with the development of the TASBE method for calibrated flow cytometry (**Beal et al.**, 2012), which builds on pre-existing calibration methods to enable high-throughput measurement of equivalent absolute units from multiple fluorescent species (often proteins, though other classes of molecule can also be used).

As an instrument, flow cytometers already fulfill two of the three assay requirements, since they break a sample into individual particles (many corresponding to individual cells) and take fluorescence measurements on multiple channels simultaneously from large numbers of those particles. Better yet, flow cytometers have become widely available, and many flow cytometers have high-throughput screening capabilities that make it easy to evaluate many samples in a short time. The measurements produced, however, are in arbitrary units, which can vary wildly depending on the machine and its settings and which are subject to drift over time. Thus, in order to transform flow cytometry into an assay capable of supporting modeling, it is necessary to add calibration controls that can enable a reliable mapping from relative to absolute units.

The TASBE method (**Beal et al.**, 2012) builds on prior methods for flow cytometry calibration (**Roederer**, 2001, 2002; **Hoffman et al.**, 2012; **Schwartz et al.**, 2004), enhancing them to allow equivalent units to be read from all channels of a flow cytometer. The TASBE method is modular and can be incorporated as an extension to nearly any other flow cytometry protocol. Cells should be prepared and gated to remove non-cell particles as dictated by the base protocol; the TASBE method just requires that each experiment also include measurements from a set of calibration controls (some of which should already be part of any experiment).

In particular, the method uses a set of four controls to compute a calibrated "color model" for converting data from arbitrary units to absolute units, as illustrated in Figure 6:

1. a negative control, to quantify autofluorescence
2. single positive controls for each fluorescent species, to quantify spectral overlap
3. fluorescent beads calibrated to an absolute standard of Molecules of Equivalent Fluorescein (MEFL), such as SpheroTech RCP-30-5A beads (**SpheroTech**, 2001))
4. for each fluorescent species not measured in the FITC channel, a multi-color control with equivalent co-expression of that species and the species measured in the FITC channel.

The first two controls are used to remove fluorescence contamination from the measurements; the latter two controls are used to convert to absolute units.

Compensation for Fluorescence Contamination Fluorescence measurements are contaminated in two ways. First, cells (and the medium in which they are suspended) have some degree of autofluorescence, adding a consistent background to any fluorescence measurement. Second, there is often overlap in the excitation and emission spectra of fluorescent species, such that the measurements for each species will include "spectral bleed" proportional to each other species' concentration and degree of overlap. The

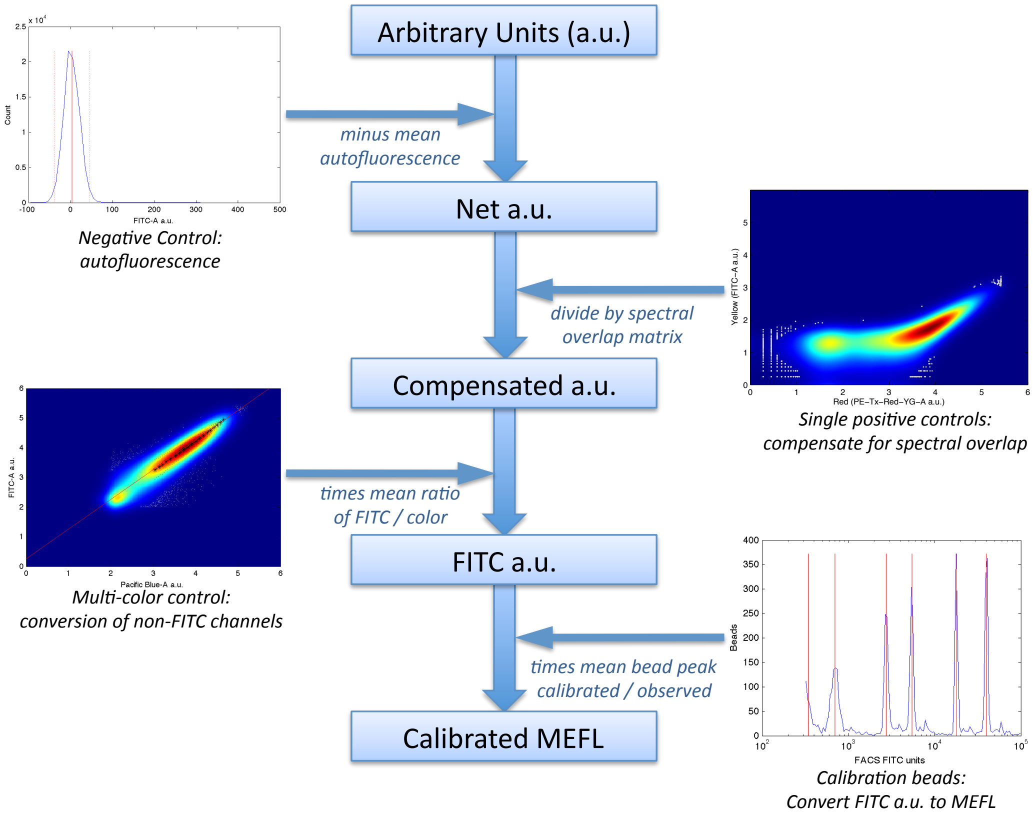
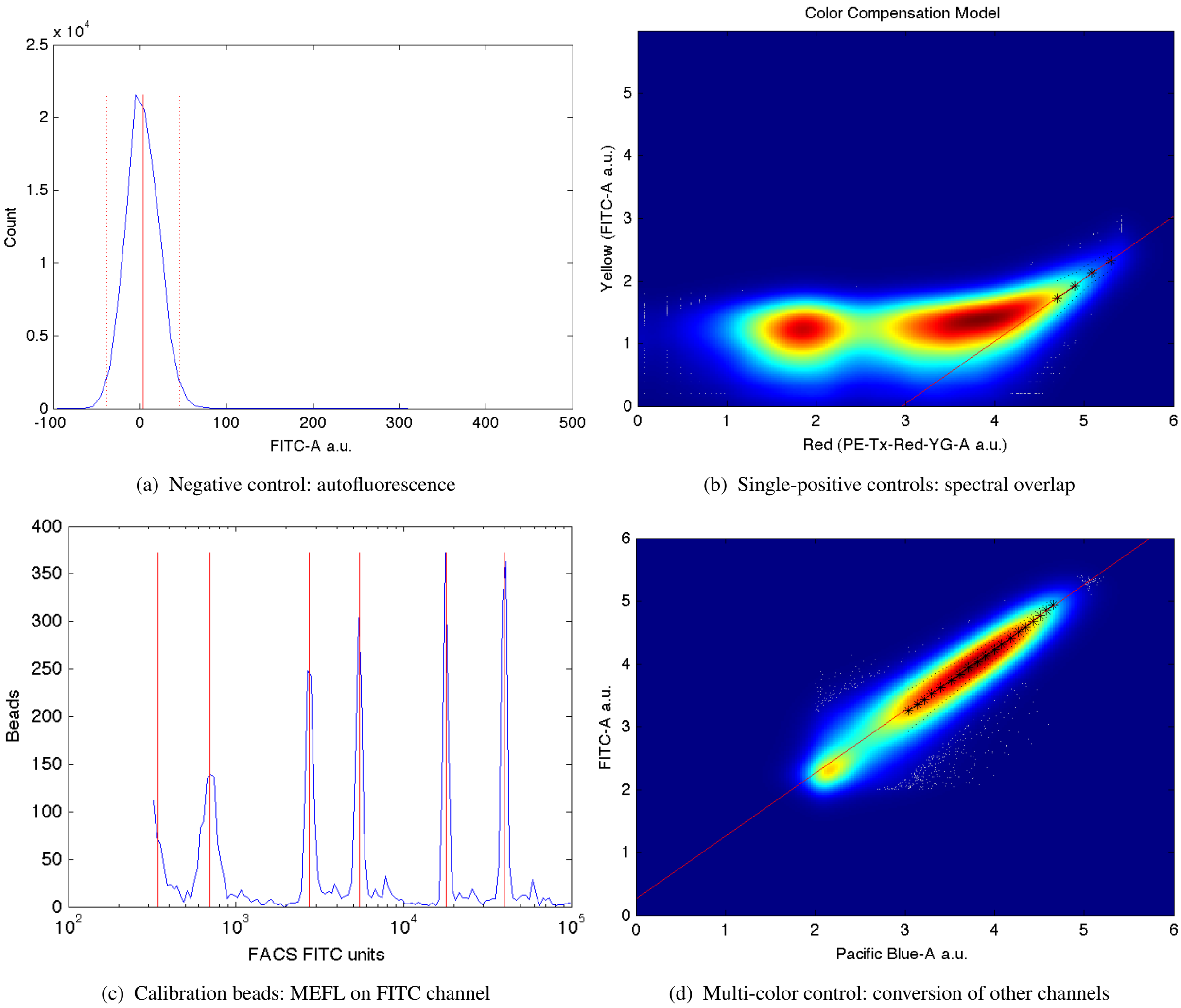

Arbitrary Units (a.u.)
minus mean autofluorescence
Net a.u.
divide by spectral overlap matrix
Compensated a.u.
times mean ratio of FITC / color
FITC a.u.
times mean bead peak calibrated / observed
Calibrated MEFL
x 10^4
Count
FITC-A a.u.
Negative Control: autofluorescence
Yellow (FITC-A a.u.)
Red (PE-Tx-Red-YG-A a.u.)
Single positive controls: compensate for spectral overlap
FITC-A a.u.
Pacific Blue-A a.u.
Multi-color control: conversion of non-FITC channels
Beads
FACS FITC units
Calibration beads: Convert FITC a.u. to MEFL

Figure 6. The TASBE method (**Beal et al.**, 2012) for calibrated flow cytometry uses four controls: correction for autofluorescence and spectral overlap is computed from the negative and single-positive controls. Calibration beads provide a conversion from arbitrary units to molecules of equivalent fluorescein (MEFL) on the FITC channel, and multi-color controls allow all other channels to be converted to equivalent FITC units and thence to MEFL. Data shown is from sample material on (**Adler et al.**, 2014).

amount of overlap depends on particular fluorescent species and the configuration and settings of the flow cytometer.

Mean autofluorescence on each channel can be estimated from a negative control, either wild-type or a null transfection/transformation (null is preferred over wild-type, because some cells' fluorescence properties change in response to the stress of transfection/transformation protocols). Autofluorescence is typically normally distributed; Figure 7(a) shows a typical example of low autofluorescence (from untransfected HEK293 cells), computing both the mean (solid red line) and two standard deviations (dotted red lines).

Once autofluorescence has been quantified, spectral overlap can be estimated from strong constitutive expression of each fluorescent protein individually. With a single protein being expressed, any fluorescence observed significantly above autofluorescence in any other channel must be the result of spectral bleed. This is a linear effect, and thus may be estimated from the mean ratio of the two measurements for highly-expressing particles after autofluorescence is removed. Figure 7(b) shows an example of computing the spectral bleed from strong constitutive expression of the red fluorescent protein mKate into the FITC channel (in this case intended to be used for quantifying EFYP expression), finding an approximate bleed of around 0.1%. Note that this is a relative measure, depending on the settings of both channels involved, rather than an absolute measure of the percentage of energy contaminating. This distinction is important because the purpose is to be able to use the (relative) measurement on one channel to correct the (relative) measurement on the other.



(a) Negative control: autofluorescence

(b) Single-positive controls: spectral overlap

(c) Calibration beads: MEFL on FITC channel

(d) Multi-color control: conversion of other channels

Figure 7. Larger versions of the sample controls shown with the TASBE method workflow in Figure 6.

Once autofluorescence and spectral bleed have been quantified, they may be removed using an affine transform, as described in (**Roederer**, 2001, 2002). High spectral bleed, however, still results in increased noise, as described in (**Roederer**, 2002). For high-precision quantification, a best-practices standard for spectral bleed is thus $< 1\%$, though higher levels can be tolerated for some purposes. The writeup in (**Beal et al.**, 2012) also describes a method for selecting species/channel combinations with minimal overlap. With presently available instrumentation and fluorescent species, it is typically possible to meet this best-practices standard for two to four species, depending on the instrument and its configuration.

Finally, note that while some flow cytometers have features to perform their own spectral compensation, it is generally better to take data uncompensated and apply compensation later. Built-in compensation is proprietary software that cannot be validated, and so it is difficult to tell whether compensation is performed correctly; moreover, in those cases where it goes wrong, it is not possible to re-compensate correctly without the original data. Similarly, at present most commercial flow cytometry software does not compensate for autofluorescence, so such compensation mechanisms should not be used in the presence of non-trivial autofluorescence.

Conversion to Absolute Units The measurements returned by a flow cytometer are highly relative and subject to change: not only do they depend on the machine, its configuration, and the laser and detector settings for a particular assay, but the instruments tend to drift in calibration over time as well. Various standard fluorescent beads have been developed to deal with this calibration problem, and have been demonstrated to provide standardizable precise measurements across a wide range of instruments and channels (**Hoffman et al.**, 2012; **Wang et al.**, 2008; **Schwartz et al.**, 2004; **Vogt Jr et al.**, 2008).

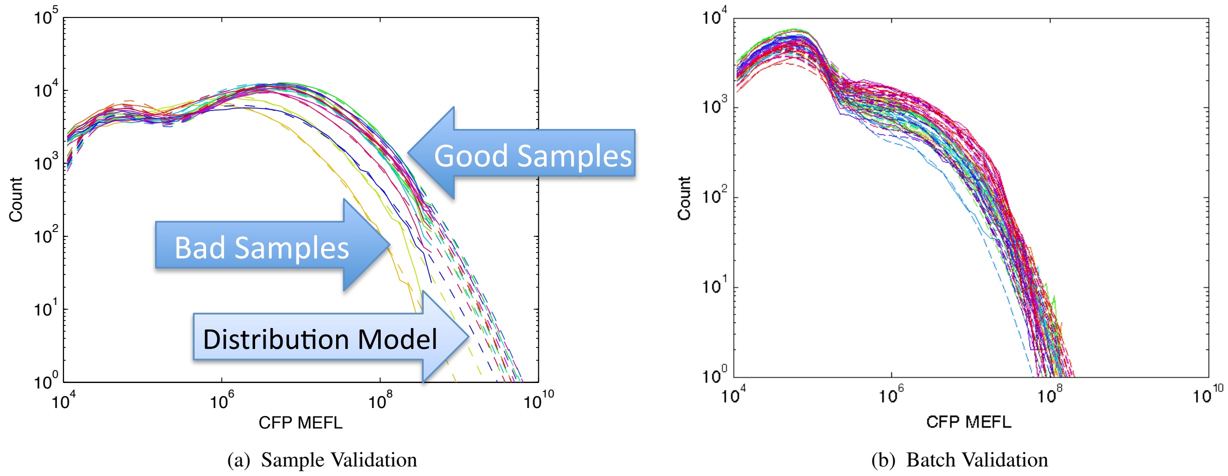
Critically, the bead manufacturer SpheroTech provides certain classes of beads (e.g., RCP-30-5A) that have been calibrated to equivalent molecules of various standard fluorescent stains (**SpheroTech**, 2001). These bead samples contain a mixture of beads with multiple distinct levels of fluorescence and non-uniform gaps between levels. This means that a linear conversion from relative units to absolute units, such as Molecules of Equivalent Fluorescein (MEFL) can be computed simply by finding the peaks in the appropriate channel of a fluorescence histogram (e.g., the FITC channel for MEFL) and matching against the list of calibration levels. Figure 7(c) shows such an example of peak identification on a sample of SpheroTech RCP-30-5A beads. Note the uneven gaps between peaks, which allow unique identification even when only a few peaks are visible.

Standard fluorophore measurements, however, do not provide comparable units between channels. Rather, each channel is characterized with respect to a different fluorescent stain and the relationship between fluorescent stains is in general different than the relationship between the various fluorescent proteins. Further, the fluorescence of various fluorescent proteins may depend on the context in which they are expressed.

The TASBE method obtains equivalent units by selecting one of the standard units (MEFL is recommended, as its greenish/yellow range is one of the most widely used channels) and computing a linear conversion factor from other channels via a multi-color control. A multi-color control must strongly constitutively express both the fluorescent protein measured in the standard channel and at least one other fluorescent protein. Each fluorescent protein in the multi-color control must be expressed using equivalent promoters and context. In some contexts, such as mammalian cells, this is relatively easy: there is little interaction between promoter and coding sequence in an expression cassette, and each expression cassette can be placed in its own plasmid and cotransfected. In bacterial cells, on the other hand, where the interaction with the 5'UTR is more significant and cotransfection typically extremely difficult, it is first necessary to validate that there is sufficient insulation between the fluorescent proteins by comparing different constructs. Figure 7(d) shows an example of finding a linear conversion factor from compensated Pacific Blue arbitrary units to compensated FITC arbitrary units in mammalian HEK293 cells, using a cotransfection of two plasmids, one expressing EBFP2, the other EFYP, both under the same strong promoter. Multiplying by the conversion factor changes Pacific Blue arbitrary units to FITC arbitrary units, which can then be converted to MEFL, thus allowing blue and yellow fluorescent proteins to be measured in equivalent absolute units.

Finally, note that it is certainly possible to go beyond measurements of equivalent fluorescence to estimates of number of molecules. In many cases, this may be desirable to do (e.g **Rosenfeld et al.** (2007)), but it is not always necessary and may introduce additional noise. It is not always necessary because the base requirement for effective modeling is absolute units, and MEFL is already such. Further, fluorescence is being measured directly, while molecule counts are inferred based on additional estimates; differences in chemical environment, quenching, and other such factors may affect the fluorescence per molecule, however, and can create distortions in molecule estimates.

Putting it all together, these four stages of the TASBE method, applied following the workflow in Figure 6, provide absolute unit measurements from large numbers of single cells. This method thereby provides an example of a measurement assay sufficient to serve as a foundation for the development of model-driven engineering methods, as will be shown in the next section.



(a) Sample Validation

(b) Batch Validation

Figure 8. Population distribution of a constitutive fluorescent protein (CFP) can be used to identify protocol problems, from individual samples (a) to entire replicates (b). Data shown is from sample material on (**Adler et al.**, 2014): solid lines are observed distribution, dashed lines are bimodal model fit, used for quantitative comparison of sample to expected distribution.

5 FROM MEASUREMENT TO PREDICTION AND DEVICE ENGINEERING

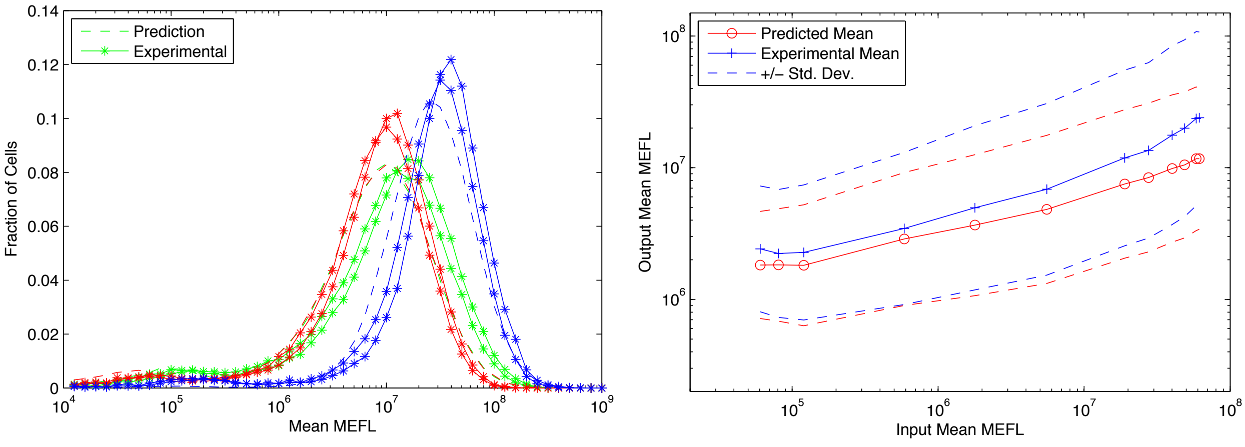
The results of calibrated large-scale per-cell measurement assays can provide a firm foundation for the development of model-driven engineering methods. This section demonstrates this relationship though presentation of three recent examples of ways in which model-driven engineering methods have been derived from calibrated flow cytometry data: improving the quality of data obtained from assays through intra-sample validation, predicting multi-component systems from models of individual components, and debugging of complex novel components.

5.1 INTRA-SAMPLE VALIDATION

One of the frequent frustrations in biological experiments is the difficulty in distinguishing between effects due to the intended subject of study versus those due to fluctuations and errors in protocol or reagents. This difficulty can happen at any scale, from individual samples to correlated sets of samples, to entire replicates or experiments. Standard controls can help to identify problems, but cannot detect problems that do not affect the control or affect it more subtlely.

With the capability to measure and compare absolute fluorescent distributions across experiments, however, it is possible to validate each individual sample using the distribution of fluorescence within the sample. This can be implemented, for any assay not expected to have a strong impact on cell viability, by including a strong constitutively expressed fluorescent protein in the test construct. Often this can even be done without modifying the system under study at all, because such a fluorescent protein is already included as a transfection marker.

Once a baseline model of variation for fluorescent distributions has been established (e.g., from single positive controls of the constitutive protein), then each sample can be validated individually by evaluating its distribution of constitutive fluorescence. Whenever there is a significant difference from baseline, it indicates either that something has varied significantly in the protocol or that there is a strong impact on cell viability (e.g., resource competition, disabling of function in expression machinery). Not only can this method detect problems with individual samples, it can also detect problems that may not be visible from population-averaged data alone. For example, contamination or sample degradation may not appear to have a significant effect on the mean, but may contain anomalous “bumps” in other portions of the distribution.



(a) Multi-Replicon Mixture

(b) Two-Repressor Cascade

Figure 9. Component models built using calibrated flow cytometry data enable high-precision predictions of multi-component systems. For example, (a) models of single- and dual-replicon transfection can predict (dashed lines) the observed histogram for distribution of fluorescence (lines with stars, two replicates) in a three-replicon mixture (figure from **Beal et al.** (2014)), and (b) repressor models can predict the observed mean and population distribution of fluorescent expression of combinational circuits such as the two-repressor cascade shown (figure from **Davidsohn et al.** (2014)).

Figure 8 shows examples of applying this method to assays of mammalian cells that have been cotransfected with both a circuit under study and a constitutive transfection marker, showing the expression histogram of the constitutive marker for each sample in the experiment in a unique color. In each case, all of the samples in an experiment are compared on a single graph and fitted against a bimodal log-normal distribution model of the expected transfection distribution, where the upper component is successfully transfected cells and the lower component is untransfected cells. For the experiments shown, the baseline model has been established as the active component containing at least 50% of the cells and having a geometric mean of approximately 10^7 MEFL. Figure 8(a) shows an experiment with mostly normal transfections and a small number of anomalous samples: the three lowest samples, whose distributions are clearly different than the rest, are rejected while the rest are retained. Figure 8(b) shows a much more extreme problem: an entire batch of data with a significantly degraded transfection efficiency.

This approach thus constitutes a model-driven engineering method, though not one directly tied to design. Rather, being able to compare assays of per-cell sample data against an absolute distribution model enables both more principled sample rejection and early detection of problems that might otherwise lead to large amounts of wasted time and effort.

5.2 PREDICTION OF MULTI-COMPONENT SYSTEMS

Distribution models can also, of course, be applied directly to system design, by using them to predict the consequences of different design choices. By allowing examination of the different modes of variation in the distribution of expression in a population of cells, calibrated flow cytometry also allows more parameters relevant to the mechanisms regulating expression to be quantified.

Consider, for example, the highly asymmetric expression distribution shown in Figure 5(b), from (**Beal et al.**, 2014). The asymmetries in the distribution correspond to separable effects of different mechanisms regulating expression from Sindbis replicons transfected into BHK-21 mammalian cells, as described in Section 4.1. Combining such cotransfection data with time-series observations of single replicon expression suffices to construct a model for the expression over time of any dosage mixture of constitutive replicons. This model, in turn, allows high-precision prediction and engineering of the distribution of fluorescent expression, evolving over time, across a wide range of multi-replicon mixtures. An example of such prediction is shown in Figure 9(a), which compares the predicted and experimentally observed distributions of fluorescent expression from a mixture of 360 ng of mVenus replicon (green), 360 ng of mKate replicon (red), and 1080 ng of EBFP2 replicon (blue), observed 50 hours after transfection into BHK-21 cells.

Another example is prediction of cotransfected circuits in HEK293 mammalian cells from characterization of individual repressors, as presented in (**Davidsohn et al.**, 2014). Here, calibrated flow cytometry was applied to produce a mixed mechanistic/phenotypic model of the action of three repressor/promoter pairs (actually synthetic hybrid activator/repressor systems with promoters activated by constitutive VP16Gal4 and repressed by one of the transcriptional regulators TAL14, TAL21, or LmrA). The distribution model is parametrized by time, by input expression as indicated by a co-expressed input fluorescent protein, and by relative circuit dosage as indicated by a constitutive fluorescent protein. Models of individual repressors may then be combined to create a model of a multi-repressor circuit, allowing high-precision prediction of both the population mean and the cell-to-cell variation of output protein expression for combinational circuits, as demonstrated in (**Davidsohn et al.**, 2014) for multiple two-repressor cascades and three-repressor feed-forward circuits, such as the TAL21-TAL14 cascade shown in Figure 9(b).

From such predictions comes the ability to to eliminate configurations that are not productive to assay, and to prioritize assays for those constructs mostly likely to prove successful, as discussed in Section 3.2, providing another example of model-driven engineering based on calibrated flow cytometry.

5.3 IMPROVING DEVICE ENGINEERING

The deeper insight enabled by distribution models can also support model-driven engineering by helping to engineer devices with less mutual constraint, which decreases the expected difficulty of discovering acceptable system configurations as discussed in Section 3.2. Here, the value of being able to compare population distributions is that different aspects of the behavior of a regulatory device can affect the structure of the distribution in different ways. For example, when a constitutive fluorescent protein is included in a high-variance cotransfection, it enables the behavior of the device to be examined as a function of the relative number of circuit copies. Since different aspects of a device's behavior scale differently with copy numbers, such sub-sample decomposition can provide deeper insight into the causes of observed behavior. For example, the degree of leakage in a repressor can be quantified by the location of the inflection point where expression rises above autofluorescence in a "minus" sample. Such forms of analysis, in turn, allow the relative importance of different performance limitations to be evaluated, enabling better focusing of engineering efforts on the limiting factors in the design of a device.

For example, (**Kiani et al.**, 2014) presents new classes of repressor devices based on the CRISPR system, which are characterized using calibrated flow cytometry. Each device comprises a number of different components, including constitutive expression of the mutant protein Cas9m, which forms a targeted repressor when it binds with gRNA expressed either directly or from introns, Gal4VP16, which drives device expression unless overridden by Cas9m, and a complex hybrid promoter including multiple targeting sites for both Cas9m and and Gal4VP16. In the early stages of designing these new devices, there was a problem with high variability with respect to gRNA sequence: some gRNAs performed very well, others inexplicably poorly. An assay with a constitutive marker revealed that the less functional devices had a much lower rate of leakage expression at the input stage, indicating that the problem lay not in the action of the repressor complex on the promoter, where it was originally believed the problem lay, but in the expression of intronic gRNA. Refocusing engineering effort on that aspect of the system led to new versions of devices with greatly improved performance, which are the ones ultimately reported in (**Kiani et al.**, 2014).

Thus, just as in the previous examples, absolute unit comparison of distributions of cell behaviors enables a more model-driven approach to engineering biological organisms. The only difference is that in this case, the set of interacting mechanisms that the methods are applied to is being conceived of by the engineers as a single complex "device."

6 DISCUSSION AND DIRECTIONS FOR FUTURE DEVELOPMENT

As the field of synthetic biology continues to expand, the problems of measurement and design are becoming increasingly pressing. This paper has developed an information-based measure that can be used to determine the relative importance of good design methods in synthetic biology applications. Applying this measure shows that precision modeling and design must play an important role in future application development. Progress in modeling and design has previously been inhibited by limitations in assay protocols that made it difficult to effectively study the distribution of expression levels within a cell population. Calibrated flow cytometry, however, is an example of a recently developed method that overcomes this limitation, and applications of this method demonstrate how comparison of expression level distributions can enable deeper insight into cell behavior as well as high-precision modeling and design.

6.1 DIRECTIONS FOR FUTURE DEVELOPMENT

The future of synthetic biology engineering rests on three complementary pillars of development: high-throughput screening, improved device families, and precision modeling and design. The first two of these are already the subjects of heavy investigation and rapid progress, while the third has proved more elusive. Recent results from calibrated flow cytometry, however, indicate that there is now a sufficient foundation for renewed investigation of precision modeling and design.

Strategic investment in this area has the potential for transformative impact across a broad space of applications for engineered biological organisms. With respect to calibrated flow cytometry in particular, there are three key directions for work:

- **Exploitation and integration of calibrated flow cytometry:** Calibrated flow cytometry is a readily accessible technology, as it builds on instruments and methods already widely in use, requiring only a few simple additional controls. When combined with more sophisticated data analysis, it has the potential to radically improve the amount of insight and precision of models that can be derived from experiments, as illustrated in Section 5. Significant impact is thus likely to be obtained from the dissemination and exploitation of calibrated flow cytometry techniques, and their integration with a wide variety of systems and synthetic biology projects.
- **Application to a broader range of organisms:** At present, calibrated flow cytometry has been applied primarily to the engineering of sensing and control circuits in mammalian cells. Preliminary work has already begun on extension to other cell types, as discussed in Section 4, each of which may require its own modifications and refinements in order to operate correctly: for example, multi-color controls are harder to calibrate in bacteria, while plant cells have extremely strong autofluorescence due to chlorophyll.
- **Application beyond sensing and control circuits:** Calibrated flow cytometry should also be applicable to problems outside of the realm of sensing and control circuits, though it will likely need to be used in combination with other assays. For example, is should be possible to examine population distributions for chemical synthesis with calibrated metabolite sensors, or to apply calibrated flow to tissue engineering by using it in combination with imaging.

Calibrated flow cytometry, of course, is just one of many potential assays for obtaining information to enable high-precision modeling and design. In fact, fluorescence is far from an ideal quantity for such assays, as it is often only a proxy measure for other, more relevant properties of cells. An important longer term goal is thus to develop new assays that can provide the same power to examine population distributions, but for other quantities such as molecule count or the configuration of sub-cellular structures.

DISCLOSURE/CONFLICT-OF-INTEREST STATEMENT

The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

AUTHOR CONTRIBUTIONS

JB is the sole author of this article.

ACKNOWLEDGEMENTS

The perspectives set forth in this review have been influenced by a number of discussions with colleagues over the last two years. Particularly useful were the discussions at the "Metrology for Mammalian Synthetic Biology" Workshop held in May 2013 at MIT.

Funding: This work has been partially supported by DARPA under grant W911NF-11-054 and contract HR0011-10-C-0168; the views and conclusions contained in this document are those of the author and not DARPA or the U.S. Government.

REFERENCES

Adler, A., Yaman, F., and Beal, J. (2014), TASBE Tools website, https://synbiotools.bbn.com/

Amdahl, G. M. (1967), Validity of the single processor approach to achieving large-scale computing capabilities, in AFIPS Conference Proceedings, volume 30, volume 30, 483–485, doi:doi:10.1145/1465482.1465560

Beal, J. and Bachrach, J. (2008), Cells are plausible targets for high-level spatial languages, in Proceedings of the 2008 Second IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshops (IEEE Computer Society, Washington, DC, USA), SASOW '08, 284–291, doi:10.1109/SASOW.2008.14

Beal, J., Lu, T., and Weiss, R. (2011), Automatic compilation from high-level biologically-oriented programming language to genetic regulatory networks, *PLoS ONE*, 6, 8, e22490, doi:10.1371/journal.pone.0022490

Beal, J., Wagner, T. E., Kitada, T., Azizgolshani, O., Parker, J. M., Densmore, D., et al. (2014), Model-driven engineering of gene expression from rna replicons, *ACS synthetic biology*

Beal, J., Weiss, R., Yaman, F., Davidsohn, N., and Adler, A. (2012), A method for fast, high-precision characterization of synthetic biology devices, Technical Report MIT-CSAIL-TR-2012-008, MIT, technical Report: MIT-CSAIL-TR-2012-008 http://hdl.handle.net/1721.1/69973

Bilitchenko, L., Liu, A., Cheung, S., Weeding, E., Xia, B., Leguia, M., et al. (2011), Eugene: A domain specific language for specifying and constraining synthetic biological parts, devices, and systems, *PLoS ONE*, 6, 4, e18882

Bonnet, J., Yin, P., Ortiz, M. E., Subsoontorn, P., and Endy, D. (2013), Amplifying genetic logic gates, *Science*, 340, 6132, 599–603

Borujeni, A. E., Channarasappa, A. S., and Salis, H. M. (2013), Translation rate is controlled by coupled trade-offs between site accessibility, selective rna unfolding and sliding at upstream standby sites, *Nucleic acids research*, gkt1139

Brustad, E. M. and Arnold, F. H. (2011), Optimizing non-natural protein function with directed evolution, *Current opinion in chemical biology*, 15, 2, 201–210

Canton, B., Labno, A., and Endy, D. (2008), Refinement and standardization of synthetic biological parts and devices, *Nature Biotechnology*, 26, 7, 787–793

Carlson, E. D., Gan, R., Hodgman, C. E., and Jewett, M. C. (2012), Cell-free protein synthesis: applications come of age, *Biotechnology advances*, 30, 5, 1185–1194

Carlson, R. H. (2011), Biology Is Technology : The Promise, Peril, and New Business of Engineering Life (Harvard University Press)

Chudakov, D. M., Matz, M. V., Lukyanov, S., and Lukyanov, K. A. (2010), Fluorescent proteins and their applications in imaging living cells and tissues, *Physiological Reviews*, 90, 3, 1103–1163

Cobb, R. E., Si, T., and Zhao, H. (2012), Directed evolution: an evolving and enabling synthetic biology tool, *Current opinion in chemical biology*, 16, 3, 285–291

Crouch, A. (1999), Design-for-test for Digital IC's and Embedded Core Systems (Prentice Hall)

Czar, M. J., Cai, Y., and Peccoud, J. (2009), Writing dna with genocad., *Nucleic Acids Research*, 37, Web Server issue, W40–W47

DallAsta, L., Ramezanpour, A., and Zecchina, R. (2008), Entropy landscape and non-gibbs solutions in constraint satisfaction problems, *Physical Review E*, 77, 3, 031118

Davidsohn, N., Beal, J., Kiani, S., Adler, A., Yaman, F., Li, Y., et al. (2014), Accurate predictions of genetic circuit behavior from part characterization and modular composition, *ACS Synthetic Biology*, doi:dx.doi.org/10.1021/sb500263b

Dietrich, J. A., McKee, A. E., and Keasling, J. D. (2010), High-throughput metabolic engineering: Advances in small-molecule screening and selection, *Annual Review of Biochemistry*, 79, 563–90, doi:0.1146/annurev-biochem-062608-095938

Duvall, P. M. (2007), Continuous Integration: Improving Software Quality and Reducing Risk., ISBN 0-321-33638-0 (Addison-Wesley)

Ellis, T., Wang, X., and Collins, J. (2009), Diversity-based, model-guided construction of synthetic gene networks with predicted functions, *Nature biotechnology*, 27, 5, 465–471

Elowitz, M. and Leibler, S. (2000), A synthetic oscillatory network of transcriptional regulators, *Nature*, 403, 6767, 335–338

Esvelt, K. M., Carlson, J. C., and Liu, D. R. (2011), A system for the continuous directed evolution of biomolecules, *Nature*, 472, 7344, 499–503

Ferber, D. (2004), Synthetic biology. microbes made to order., *Science (New York, NY)*, 303, 5655, 158

Forrest, S. and Mitchell, M. (1993), What makes a problem hard for a genetic algorithm? some anomalous results and their explanation, *Machine Learning*, 13, 2-3, 285–319

Galdzicki, M., Wilson, M. L., Rodriguez, C. A., Pocock, M. R., Oberortner, E., Adam, L., et al. (2012), Synthetic Biology Open Language (SBOL) Version 1.1.0, RFC 87

Gardner, T. S., Cantor, C. R., and Collins, J. J. (2000), Construction of a genetic toggle switch in *escherichia coli*, *Nature*, 403, 339—342

Gulati, S., Rouilly, V., Niu, X., Chappell, J., Kitney, R. I., Edel, J. B., et al. (2009), Opportunities for microfluidic technologies in synthetic biology, *Journal of The Royal Society Interface*, 6, Suppl 4, S493–S506

Hasty, J., McMillen, D., and Collins, J. J. (2002), Engineered gene circuits, *Nature*, 420, 6912, 224–230

Hillson, N. J., Rosengarten, R., and Keasling, J. D. (2012), j5 dna assembly design automation software, *ACS Synthetic Biology*, 1, 1

Hoffman, R. A., Wang, L., Bigos, M., and Nolan, J. P. (2012), Nist/isac standardization study: Variability in assignment of intensity values to fluorescence standard beads and in cross calibration of standard beads to hard dyed beads, *Cytometry Part A*, 81, 9, 785–796

Hogg, T., Huberman, B. A., and Williams, C. P. (1996), Phase transitions and the search problem, *Artificial intelligence*, 81, 1, 1–15

Huynh, L., Tsoukalas, A., Kppe, M., and Tagkopoulos, I. (2013), Sbrome: A scalable optimization and module matching framework for automated biosystems design, *ACS synthetic biology*, 2, 5, 263–273

INCOSE (2010), Systems Engineering Handbook - A Guide for System Life Cycle Processes and Activities, Version 3.2 (International Council On Systems Engineering (INCOSE))

Kanefsky, B. and Taylor, W. (1991), Where the really hard problems are, in Proceedings of IJCAI, volume 91, volume 91, 163–169

Kelly, J. R., Rubin, A. J., Davis, J. H., Ajo-Franklin, C. M., Cumbers, J., Czar, M. J., et al. (2009), Measuring the activity of biobrick promoters using an in vivo reference standard, *Journal of Biological Engineering*, 3, 4

Kiani, S., Beal, J., Ebrahimkhani, M. R., Huh, J., Hall, R. N., Xie, Z., et al. (2014), Crispr transcriptional repression devices and layered circuits in mammalian cells, *Nature methods*

Knight, T. (2003), Idempotent vector design for standard assembly of biobricks, Technical Report MIT Synthetic Biology Working Group, 0, MIT CSAIL

Knight, T. F. and Sussman, G. J. (1998), Cellular gate technology, in In First International Conference on Unconventional Models of Computation (UMC98, 1–17

Kong, D. S., Carr, P. A., Chen, L., Zhang, S., and Jacobson, J. M. (2007), Parallel gene synthesis in a microfluidic device, *Nucleic acids research*, 35, 8, e61

Kosuri, S., Goodman, D. B., Cambray, G., Mutalik, V. K., Gao, Y., Arkin, A. P., et al. (2013), Composability of regulatory sequences controlling transcription and translation in escherichia coli, *Proceedings of the National Academy of Sciences*, 110, 34, 14024–14029

Krzakala, F. and Kurchan, J. (2007), Landscape analysis of constraint satisfaction problems, *Physical Review E*, 76, 2, 021122

Kwok, R. (2010), Five hard truths for synthetic biology, *Nature*, 463, 7279, 288–90, doi:10.1038/463288a

Larman, C. (2004), Agile and Iterative Development: A Manager's Guide, ISBN 978-0-13-111155-4 (Addison-Wesley)

Linshiz, G., Stawski, N., Poust, S., Bi, C., Keasling, J. D., and Hillson, N. J. (2012), Par-par laboratory automation platform, *ACS synthetic biology*, 2, 5, 216–222

Liu, C. C., Qi, L., Lucks, J. B., Segall-Shapiro, T. H., Wang, D., Mutalik, V. K., et al. (2012), An adaptor from translational to transcriptional control enables predictable assembly of complex regulation, *Nature methods*, 9, 11, 1088–1094

Lou, C., Stanton, B., Chen, Y.-J., Munsky, B., and Voigt, C. A. (2012), Ribozyme-based insulator parts buffer synthetic circuits from genetic context, *Nature biotechnology*, 30, 11, 1137–1142

Lux, M. W., Bramlett, B. W., Ball, D. A., and Peccoud, J. (2012), Genetic design automation: engineering fantasy or scientific renewal?, *Trends in biotechnology*, 30, 2, 120–126

Lynch, S. A. and Gill, R. T. (2012), Synthetic biology: New strategies for directing design, *Metabolic engineering*, 14, 3, 205–211

Marchisio, M. A. and Stelling, J. (2011), Automatic design of digital synthetic gene circuits, *PLoS Comput Biol*, 7, 2, e1001083, doi:10.1371/journal.pcbi.1001083

Moriya, Y., Shigemizu, D., Hattori, M., Tokimatsu, T., Kotera, M., Goto, S., et al. (2010), Pathpred: an enzyme-catalyzed metabolic pathway prediction server, *Nucleic acids research*, gkq318

Mutalik, V. K., Guimaraes, J. C., Cambray, G., Lam, C., Christoffersen, M. J., Mai, Q.-A., et al. (2013), Precise and reliable gene expression via standard transcription and translation initiation elements, *Nature methods*, 10, 4, 354–360

Myers, C. J. (2013), Platforms for genetic design automation, *Microbial Synthetic Biology*, 40, 177–202

Neumann, H., Wang, K., Davis, L., Garcia-Alai, M., and Chin, J. W. (2010), Encoding multiple unnatural amino acids via evolution of a quadruplet-decoding ribosome, *Nature*, 464, 7287, 441–444

ONeill, M., Vanneschi, L., Gustafson, S., and Banzhaf, W. (2010), Open issues in genetic programming, *Genetic Programming and Evolvable Machines*, 11, 3-4, 339–363

Pedersen, M. and Phillips, A. (2009), Towards programming languages for genetic engineering of living cells, *Journal of the Royal Society Interface the Royal Society*, 6, 4, S437–S450

Pharkya, P., Burgard, A. P., and Maranas, C. D. (2004), Optstrain: a computational framework for redesign of microbial production systems, *Genome research*, 14, 11, 2367–2376

Rodrigo, G., Carrera, J., and Jaramillo, A. (2011), Computational design of synthetic regulatory networks from a genetic library to characterize the designability of dynamical behaviors, *Nucleic Acids Res.*, 39, e138

Roederer, M. (2001), Spectral compensation for flow cytometry: visualization artifacts, limitations, and caveats, *Cytometry*, 45, 3, 194–205

Roederer, M. (2002), Compensation in flow cytometry, *Current Protocols in Cytometry*, 1–14

Rosenfeld, N., Young, J., Alon, U., Swain, P., and Elowitz, M. (2007), Accurate prediction of gene feedback circuit behavior from component properties, *Molecular Systems Biology*, 13, 3, 143

Rosenfeld, N., Young, J. W., Alon, U., Swain, P. S., and Elowitz, M. B. (2005), Gene regulation at the single-cell level, *Science*, 307, 5717, 1962–1965

Russell, S. J. and Norvig, P. (2003), Artificial Intelligence: A Modern Approach (Pearson Education), 2 edition

Salis, H. M., Mirsky, E. A., and Voigt, C. A. (2009), Automated design of synthetic ribosome binding sites to control protein expression., *Nature Biotechnology*, 27, 10, 946–950

Schmidt, M. (2010), Xenobiology: a new form of life as the ultimate biosafety tool, *Bioessays*, 32, 4, 322–331

Schwartz, A., Gaigalas, A. K., Wang, L., Marti, G. E., Vogt, R. F., and Fernandez-Repollet, E. (2004), Formalization of the mesf unit of fluorescence intensity, *Cytometry Part B: Clinical Cytometry*, 57, 1, 1–6

SpheroTech (2001), Measuring molecules of equivalent fluorescein (mefl), pe (mepe) and rpe-cy5 (mepcy) using sphero rainbow calibration particles, Technical Report SpheroTechnical Notes: STN-9, Rev C 071398, SpheroTech

Stanton, B., Nielsen, A., Tamsir, A., Clancy, K., Peterson, T., and Voigt, C. (2014), Genomic mining of prokaryotic repressors for orthogonal logic gates, *Nature Chemical Biology*, 10, 2, 99–105, doi:10.1038/nchembio.1411

Stricker, J., Cookson, S., Bennett, M. R., Mather, W. H., Tsimring, L. S., and Hasty, J. (2008), A fast, robust and tunable synthetic gene oscillator, *Nature*, 456, 7221, 516–519

Sun, Z. Z., Hayes, C. A., Shin, J., Caschera, F., Murray, R. M., and Noireaux, V. (2013), Protocols for implementing an escherichia coli based tx-tl cell-free expression system for synthetic biology, *Journal of visualized experiments: JoVE*, , 79, doi:10.3791/50762

Tabor, J. J., Salis, H. M., Simpson, Z. B., Chevalier, A. A., Levskaya, A., Marcotte, E. M., et al. (2009), A synthetic genetic edge detection program, *Cell*, 137, 7, 1272–1281

Vasilev, V., Liu, C., Haddock, T., Bhatia, S., Adler, A., Yaman, F., et al. (2011), A software stack for specication and robotic execution of protocols for synthetic biological engineering, *3rd International Workshop on Bio-Design Automation*

Vogt Jr, R. F., Marti, G. E., and Zenger, V. (2008), Quantitative fluorescence calibration: a tool for assessing the quality of data obtained by fluorescence measurements, in Standardization and Quality Assurance in Fluorescence Measurements I (Springer), 3–31

Wang, L., Gaigalas, A. K., Marti, G., Abbasi, F., and Hoffman, R. A. (2008), Toward quantitative fluorescence measurements with multicolor flow cytometry, *Cytometry Part A*, 73, 4, 279–288

Weiss, R. (2001), Cellular Computation and Communications using Engineered Genetic Regulatory Networks, Ph.D. thesis, MIT

Yaman, F., Bhatia, S., Adler, A., Densmore, D., and Beal, J. (2012), Automated selection of synthetic biology parts for genetic regulatory networks, *ACS Synthetic Biology*, 1, 8, 332–344, doi:10.1021/sb300032y

Yousofshahi, M., Lee, K., and Hassoun, S. (2011), Probabilistic pathway construction, *Metabolic engineering*, 13, 4, 435–444

Zdeborová, L. (2008), Statistical physics of hard optimization problems, *arXiv preprint arXiv:0806.4112*