Metabolic Engineering Communications 2 (2015) 30–38

Contents lists available at ScienceDirect

Metabolic Engineering Communications

journal homepage: www.elsevier.com/locate/mec

The LASER database: Formalizing design rules for metabolic engineering

James D. Winkler, Andrea L. Halweg-Edwards, Ryan T. Gill*

Department of Chemical and Biological Engineering, University of Colorado-Boulder, Jennie Smoly Caruthers Biotechnology Building, Research Park, Boulder, CO 80303, USA

ARTICLE INFO

Article history:
Received 17 April 2015
Accepted 5 June 2015
Available online 16 June 2015

Keywords:
Metabolic engineering
Synthetic biology
Standardization
Design tools

ABSTRACT

The ability of metabolic engineers to conceptualize, implement, and evaluate strain designs has dramatically increased in the last decade. Unlike other engineering fields, no centralized, open-access, and easily searched repository exists for cataloging these designs and the lessons learned from their construction and evaluation. To address this issue, we have developed a repository for metabolic engineering strain designs, known as LASER (Learning Assisted Strain EngineeRing, laser.colorado.edu) and a formal standard for disseminating designs to metabolic engineers. Curation of every available genetically-defined *E. coli* and *S. cerevisiae* strain from 310 metabolic engineering papers published over the last 21 years yields a total of 417 designs containing a total of 2661 genetic modifications. This collection has been deposited in LASER and represents the known bibliome of genetically defined and tested metabolic engineering designs in the academic literature. Properties of LASER designs and the analysis pipeline are examined to provide insight into LASER capabilities. Several future research directions utilizing LASER capabilities are discussed to highlight the potential of the LASER database for metabolic engineering.

1. Introduction

In recent years, metabolic engineers have made substantial progress in the engineering of cellular biofactories (Woolston et al., 2013) that address critical economic and health needs, such as the production of fuels and chemicals (Gronenberg et al., 2013; Jones et al., 2015), pharmaceuticals (Nielsen, 2013; Marienhagen and Bott, 2013; Martin et al., 2003), and many other compounds (Keasling, 2010). These efforts have been supported by a multitude of recent experimental and computational advances (Fisher et al., 2014). However, similar advances in the storage and analysis of these innovative designs have not been forthcoming, making it increasingly difficult to build upon these successful examples to further improve yields, strain robustness, and other characteristics. Given the continuing expansion of the metabolic engineering literature (Fig. 1), storage and analysis of published designs is more important than ever. Previous efforts to impose standardization on genetic part and design information have attempted to tackle this issue in a variety of ways (Endy et al., 2005; Andrianantoandro et al., 2006; Galdzicki et al., 2014; Bilitchenko et al., 2011). Formalizing key facets of design will allow metabolic engineers to better harness past experience to develop novel, improved biocatalysts for the future (Canton et al., 2008) and to enable the "science" of metabolic engineering (Bailey, 1991).

Rule and knowledge codification is an integral part of more traditional engineering fields (e.g. electrical, mechanical, and civil engineering). Engineers in these fields routinely make use of extensive databases of standardized process units, empirical and theoretical design laws, and other data that help newcomers and experts alike to use the current best design practices in their field (Sinnott, 2009). Having codified design rules, such as removing corrosives from process units quickly, and empirical laws to guide design allows for the development of fully automated process design software that has massively increased engineering productivity and reliability. These resources both reduce human error (as engineers adhering to a relatively fixed process flow are less likely to miss critical errors in their designs) and improve productivity, as there is less need to re-discover previous engineering innovations that were discovered empirically. Chemical engineers have developed comprehensive bodies of empirical relations to describe critical fluid dynamics, thermodynamics, reaction kinetics, and other natural phenomenon that critically influence unit operations (Green and Perry, 2007), in addition to formalized design methodologies to minimize human error (Towler and Sinnott, 2013). Mechanical, civil, and electrical engineers have done much the

* Corresponding author.
E-mail addresses: james.winkler@colorado.edu (J.D. Winkler), andrea.edwards@colorado.edu (A.L. Halweg-Edwards), rtg@colorado.edu (R.T. Gill).

http://dx.doi.org/10.1016/j.meteno.2015.06.003

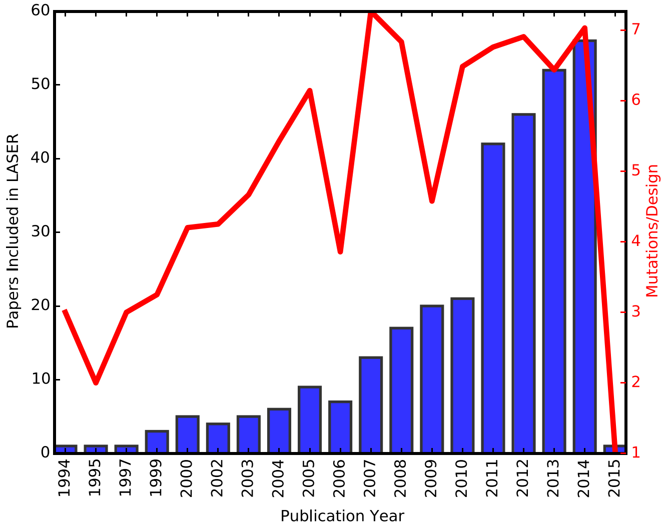

Fig. 1. Publications captured over time alongside the number of mutations per design. The majority of curated papers were published between 2008 and 2014. Papers published earlier tend to include less information about the precise genotypes of their designs due to the comparatively higher cost of genome sequencing or other forms of strain analysis. The average number of mutations per designs appears to increase fairly steadily over time (excluding the small sample size for designs published in 2015).

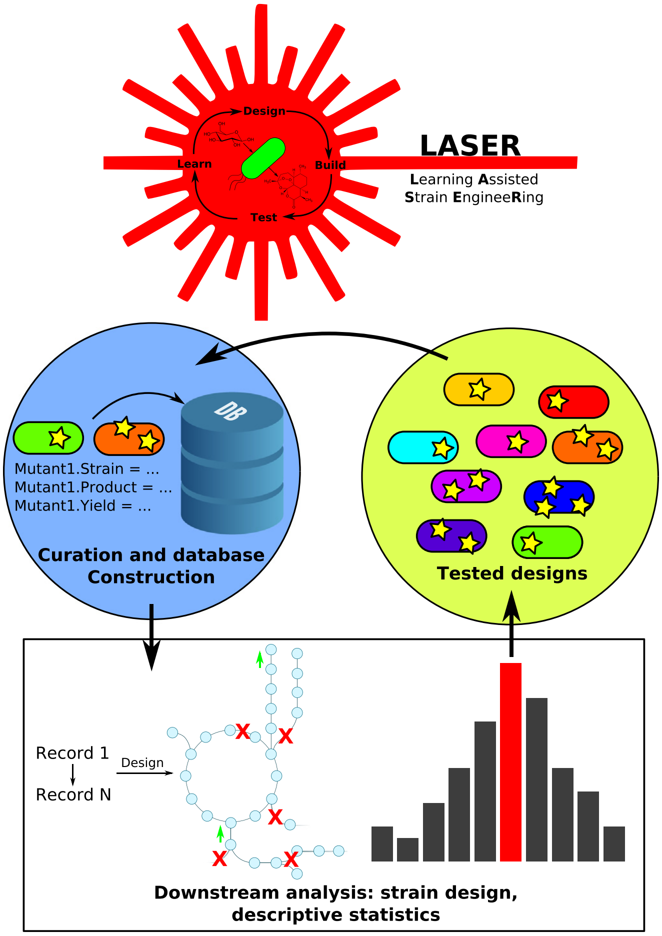

Fig. 2. The essential LASER analysis cycle. First, tested (experimentally validated) designs are deposited into LASER, followed by any necessary curation steps to standardize the model data representation. These data are then fed into the LASER analysis pipeline, where their mutations are implemented into the corresponding metabolic and regulatory models to enable various types of downstream analyses. Design data, such as probable chassis strains and potentially advantageous novel combinations of gene manipulations are then used to generate a new set of designs for testing.

same, with the latter group focusing on reducing errors that endanger human life or property resulting from the hazards uniquely associated with each field (Shigley, 2011; Choi, 2004). In all of these cases, these approaches and underlying software are based on decades of physical modeling, process design experience, and failures that have been codified over the years, something that is lacking in the metabolic engineering field.

Software engineering, which is broadly similar to metabolic engineering in terms of its current rapidly changing state and relatively nascent attempts at imposing standards on development practices (Shaw and Garlan, 1996), now includes a range of intricate design philosophies with various goals: reducing number of bugs per kiloline of code (Mills and Linger, 1991), enabling rapid shifts in features (Highsmith and Cockburn, 2001), or real-time responsiveness and extreme reliability for critical applications (Bush, 1990). Programming languages have also been designed to minimize or eliminate certain practices that result in frequent implementation errors, such as direct memory manipulation (Gosling and McGilton, 1995) or race conditions in parallel computing (Armstrong, 2007). While efforts to improve software engineering practices are still on-going, they have reduced the frequency of errors with substantial impacts on safety and profitability (Leveson and Turner, 1993; Easley et al., 2011). Pursuing the same type of metabolic engineering methodology and tool development to better incorporate empirical design rules and reduce errors promises to increase strain performance and experimental reproducibility.

Despite these past examples of standardization improving engineering practices, metabolic engineers have not yet benefitted from similar developments in our field. While there are powerful enzyme, species, and part databases, like those developed under the Biocyc aegis (Caspi et al., 2014), SGD (Costanzo et al., 2014), BRENDA (Schomburg et al., 2013), and others (Ham et al., 2012; Hesselman et al., 2012), these databases were not originally meant to encapsulate this type of metabolic engineering design information. The Biocyc databases, for example, contain a plethora of gene and pathway information; generally focused on the effect of individual gene mutations. BRENDA is principally concerned with enzyme properties, rather than the chassis strains hosting these enzymes. Other databases focus on different aspects of strain engineering, but similarly lack an overall emphasis on the tested designs. The fundamental issue with attempting to collate ME designs from these sources is that the desired data (mutations, products of interest, growth conditions, etc.) are scattered among many, disparately organized databases, or the data was not ever recorded. A database explicitly constructed to contain this information would therefore expedite storage and analysis of ME designs, avoiding the problem of adapting already existing specialist databases for entirely new purposes.

To address the dual problems of standardization and access that metabolic engineers currently face, we present the LASER (Learning Assisted Strain EngineeRing) database, a publicly-accessible, consistent platform for recording ME strain designs and supporting strain engineering (Fig. 2). In this paper, we introduce the concept of a formal metabolic engineering design, data sources used to construct LASER, and analyses of design properties. The public user interface for LASER is also described in detail and available at laser.colorado.edu. The discussion subsequently examines three possible downstream applications for LASER as applied to strain design that will be explored in future work.

2. Methods and materials

2.1. Implementation details

The LASER database and pipeline are designed to permit extensibility and interoperability with other software packages to the maximum extent possible. LASER records are stored as plain, human-readable XML or key-value files. All software tools are currently implemented in Python (version 2.7, python.org), and the web-server used for the public-facing user interface is based on Tornado (version 4.1, tornadoweb.org). Cobrapy, a Python implementation of the COBRA Toolbox (Ebrahim et al., 2013), is currently used for the manipulation of metabolic models in the LASER matching pipeline. Biocyc flat-files (Caspi et al., 2014) were downloaded and converted into queryable PostgresSQL databases for the matching pipeline as well. Visualizations were generated using Matplotlib (Hunter, 2007) and Escher (http://escher.github.io/). All calculations were performed on a Thinkpad T61 running Windows 7 with a Core Duo 2.2 GHz processor and 3 GB RAM.

2.2. Data sources and curation

All designs deposited in LASER so far have been curated from the peer-reviewed literature published in 43 different journals. The majority of curated designs were published recently (2008–2014) with a gradually increasing number of mutations per design over time (Fig. 1). This trend may reflect the power of inexpensive DNA synthesis combined with improved genome engineering techniques (Doudna and Charpentier, 2014; Pál et al., 2014; Warner et al., 2010; Lynch et al., 2007; Zeitoun et al., 2015). Only *Escherichia coli* or *Saccharomyces cerevisiae* chassis strains are included due to the availability of the corresponding metabolic and regulatory models; however, there is no technical limitation in LASER preventing the addition of designs based on other species, and other designs from other common metabolic engineering platforms (e.g. *Corynebacterium glutamicum*) will be added in the future. The curation process was entirely manual; automation via natural language processing was considered but not yet pursued given the heterogeneity of data representation in the ME literature. Given the continuing advancements in NLP (Cambria and White, 2014), it is possible that automatic curation could augment or entirely supplant manual data extraction in the future. Strain patents may also serve as an additional source of designs due to their inclusion of focused detail concerning genetic modifications and performance as well. We are currently evaluating ways to crowd-source data entry to expand the LASER database more rapidly.

In order to enforce standardization to the maximum extent possible, data entry for LASER is performed using the web interface described in Section 2.3 with client-side validation of user inputs. Depending on the number of designs and mutations involved, and how the paper is organized, curating a paper requires between 20 and 45 min. One of the principal challenges in analyzing LASER data is the lack of consistent representations of genes, media types, yield and titer values, and other design characteristics presented in the studies themselves. Metabolic and regulatory models also tend to use different naming schemes for their genes, along with their associated reactions and metabolites, so substantial effort has been dedicated to building a software pipeline capable of harmonizing these inconsistencies (see Section 3.3). Some manual re-curation is often required to convert author-supplied gene and target molecule names into version that can be identified in the corresponding Biocyc database or cellular model. The required additional effort is minimal, in most cases, and can often be performed en masse due to consistent usage of non-systematic gene and product names. Sequences for parts (promoters, CDSs, and so on) are not included in LASER, but will be extracted from part databases using the Synthetic Biology Open Language (SBOL) in the future (Galdzicki et al., 2014). Domain specific languages such as Eugene (Bilitchenko et al., 2011) may be especially useful for expanding LASER designs to include precise descriptions of their constituent parts.

2.3. Accessing LASER

The LASER database is publically accessible at laser.colorado.edu. Currently, the web app is equipped with two major features that allow users to submit their own designs using the data entry interface and perform simple queries to identify designs that have specific characteristics, such as the use of a particular feedstock or engineering methodology. Designs will be assigned a unique accession code to facilitate their identification and sharing in the future. Users can contribute designs to LASER through the “deposit” link provided on the navigation pane on the left side of the home page. A short form opens, and users are prompted to enter required information about the referenced article or patent, project metadata including the project difficulty as well as high-level design strategy (e.g. ad hoc design proposal or exhaustive model proposal/testing), and the specific strain design characteristics. General information includes the paper title, digital object identifier (DOI), publication year, corresponding author, publication journal, and a keyword identifier describing the project goal. Information regarding the perceived difficulty and project management styles are also requested in order for LASER developers to crowd-source information collection that may be useful for calculating project complexity. The user is required to enter specific information on each engineered organism being reported, allowing experimental reproducibility between labs as well as improved accuracy of prediction algorithms. Strain specific information includes but is not limited to details pertaining to growth media, oxygenation state, culturing system, and method for creating the mutation. The database search function allows users to access any reference in LASER that is attached to query key words. For example, the user can select to recover all records that contain the exact match to the keyword ‘anaerobic’ in the associated ‘oxygenation state’ table of the database. The number of return records can be capped at an arbitrary threshold as determined by the user, and the search can be generalized.

3. Results

3.1. Defining a metabolic engineering design

We have developed the LASER schema to address storage, exchange, and analysis issues and to establish a formal representation of a ME design (Fig. 3). A design is divided into four hierarchical categories: the paper level, containing information about the original study and its design methodology; the mutant level, detailing the host strain, the techniques used for genetic engineering, growth conditions, target metabolite, production levels, and the number of mutations; the mutation level, including the modified or inserted gene and their genetic modifications, followed by the annotation level, including information such as RBS sequences, promoters, and plasmid copy numbers. There is no limit on the number of mutations per design, or on the number of designs per paper, patent or other unit of published work, although most papers only contain two strain designs on average, each harboring approximately seven mutations. Quantitative statistics concerning several design properties are presented in Section 3.2.

The complete list of descriptors for each level is shown in

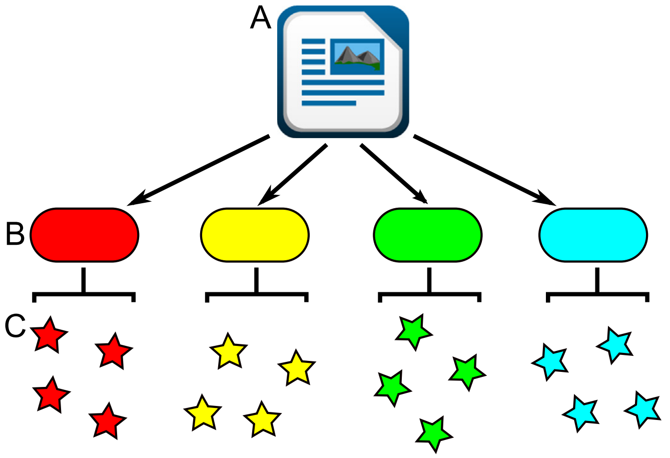

Fig. 3. The anatomy of a LASER record storing a metabolic engineering design. (A) General features concerning the study are stored at the paper level, while the mutant level (B) contains information about the strain used, growth conditions, yield and titer information, and the engineering approach. The mutation level (C) focuses on how and why specific genes are mutated, along with any necessary annotations.

Supplementary Table 1; we captured all of the information needed for experimental reproduction, balanced with the need for simple and accurate data curation. The complete definition is formalized using a standard XML schema (Supplementary File 1) that also enumerates the allowable data types and entries for each data field to enforce consistency for supplied textual data, and to permit additions to the schema in light of new developments in the metabolic engineering field. In addition to specifying the “what” of a metabolic engineering design, the LASER XML schema also permits users to write and read LASER designs without the need for specialized software tools beyond commonly available XML parsers. Text editors can also be used to edit LASER records directly, as they are stored in a human-readable format. All LASER records are stored in this format to simplify their distribution and analysis.

3.2. Properties of LASER designs

A total of 417 designs from 310 papers are currently contained within LASER; Table 1 summarizes the key statistics for the current iteration of the database. Published papers contain an average of two designs ($\mu = 1.35$, $\sigma = 1.06$), with each design harboring seven modifications ($\mu = 6.34$, $\sigma = 4.39$). Overexpression, plasmid cloning, deletion, and genomic integration of genes are by far the most common implemented modifications (Fig. 6). As tools for manipulating genomes continue to expand (e.g. recombineering and CRISPR), it is expected that this distribution will change.

LASER records also contain substantial data regarding design methodologies, and so it is also possible to broadly survey the design techniques employed by the ME field (Fig. 5). Despite the increasing use of computational methods (FBA Antoniewicz, 2015 and elementary mode analysis Trinh et al., 2009), the use of human intuition remains the most common approach to defining and implementing ME designs. This result may indicate that metabolic engineers find these tools somewhat difficult to use or too computationally intensive for their projects, or believe that inferences derived from their past experiences are more effective in producing the desired level of strain performance. Other approaches relying on random evolution or genome-wide forward-engineering libraries comprise a relatively small proportion of the LASER database, but they have also revealed unexpected modifications to increase production or growth (Alper et al., 2005; Hayashi and Tabata, 2013) that cannot be detected using purely rational approaches. Continuing interest in using these techniques for enhancing strain traits (Lynch et al., 2007; Warner et al., 2010; Winkler and Kao, 2012) indicates that their proportion of applied methodologies may increase in the future. Perhaps not surprisingly, protein engineering is performed infrequently despite its potential utility in circumventing issues with enzyme performance and specificity, most likely due to the difficulties in rational protein design (Frushicheva et al., 2014; Khoury et al., 2014) and the need to laboriously screen mutant libraries. We expect this distribution to change dramatically in the future as usability enhancements for various computational and experimental tools along with increased automation are developed.

Table 1
LASER summary statistics.

Parameter	Value	Comments
Records	310	Curated from literature
Designs	417	279 *E. coli*, 138 yeast
Journals	43	Includes papers with genotyped strains
Project methodologies	7	Design approaches for entire project
Mutant design methods	38	Design approaches for strain engineering
Gene sources	216	Sources of heterologous genes
Products	149	Number of unique target metabolites

Basic statistics concerning designs deposited in LASER.

Visualizing the complete set of metabolic alterations in *E. coli* (Fig. 4) reveals that mutations predominantly affect only a few main pathways in the strain, which makes sense given the relatively small number of metabolic branch points that control flux for the synthesis of various metabolites of interest. Manipulation of the pentose phosphate pathway is also common, most likely as an attempt to modulate levels of NADPH in the cell for redox balancing. Although this map only includes modifications to the native metabolic network, certain types of heterologous modifications are also common, such as expression and modulation of the mevalonate pathway genes in *E. coli*. The frequent alterations of these pathways suggest that core designs could be standardized and easily reused.

3.3. The LASER analysis pipeline

Due to the standardized LASER design representation, software tools can be developed to automatically implement the genetic modifications specified in a LASER record in the corresponding metabolic (COBRA Schellenberger et al., 2011; Ebrahim et al., 2013) or regulatory model (Salgado et al., 2013; Teixeira et al., 2014). Comprehensive *E. coli* and yeast metabolic models are used in this pipeline (Orth et al., 2011; Heavner et al., 2012). This process is conceptually simpler than other types of automated model generation that have been explored in the literature (Henry et al., 2010), since the base models for the organisms in question already exist. We have made substantial progress in developing an entirely automated pipeline, as depicted in Fig. 7 for generating modified cellular (metabolic and regulatory) models using LASER records and various Biocyc databases. The essential process involves identification of the systematic name for a LASER entry, its corresponding reactions, and the addition of the gene, reactions, and metabolites to the final model. Once this pipeline is publicly introduced, users will be presented with a report detailing the model generation results, including any failures to associate LASER entries to the model or Biocyc entries. Currently, only mutations resulting in deletion, repression, and overexpression are implemented by the pipeline due to the difficulty in representing more complex types of genetic alterations in these simplified cellular models. Descriptive statistics identifying both problematic genes and reactions can be generated by processing the entirety of LASER through this pipeline, a process that requires approximately

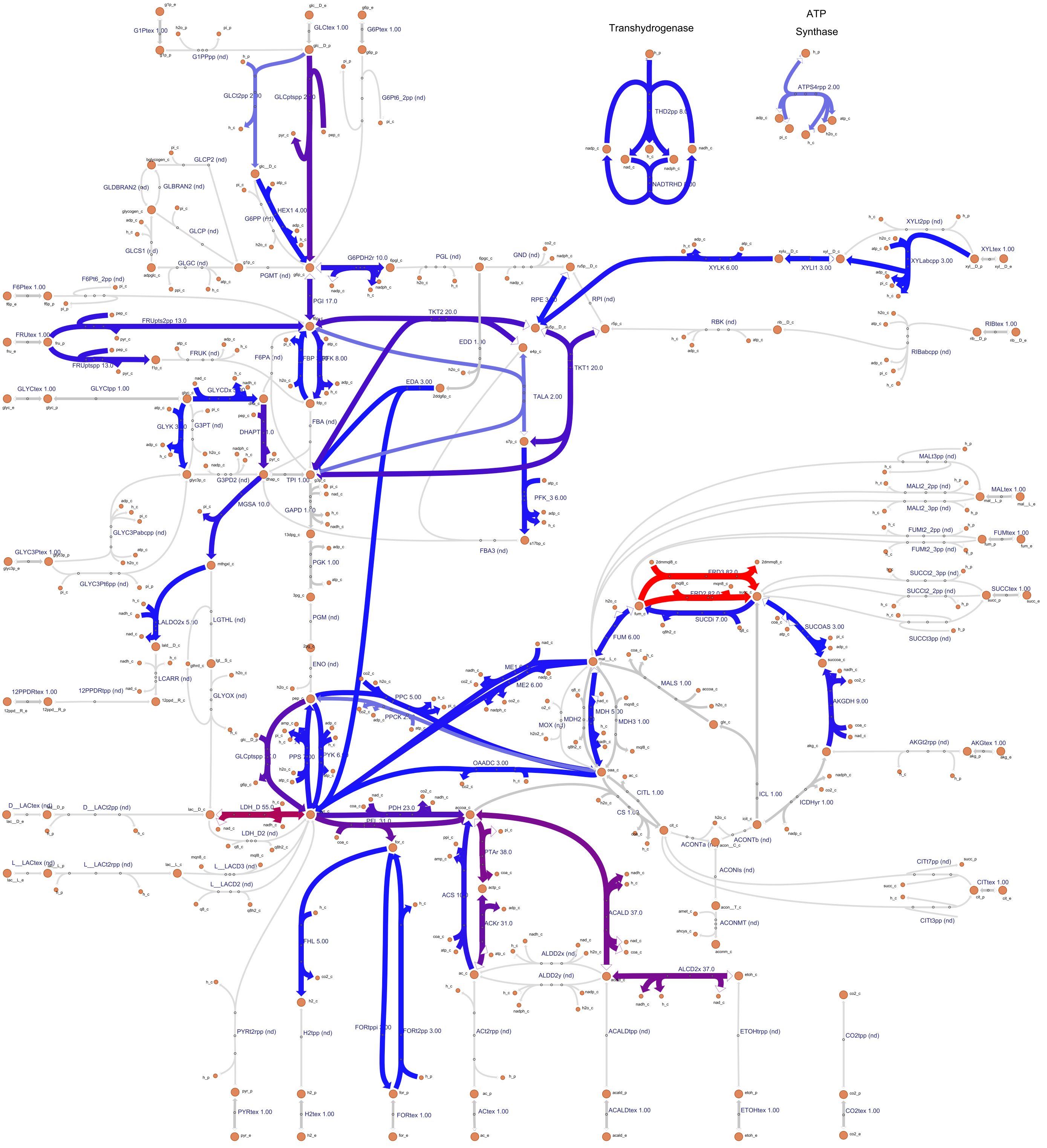

Fig. 4. Map of iJO1366 central metabolism with frequently modified reactions highlighted. Less commonly manipulated reactions are colored in blue, while reactions in purple and red are more frequently altered. Manipulated genes not involved in central carbon metabolism are not included. This visualization was produced using Escher (http://escher.github.io/). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

1 h on inexpensive hardware.

The analysis pipeline is relatively accurate for both metabolic and regulatory model generation. Of the 2601 (non-unique) mutated genes in the entire set of LASER records, 2116 can be mutated in the corresponding metabolic model (80% success rate) resulting in changes to a total of 4700 metabolite or reactions nodes. The remaining 485 genes either cannot be associated with a standardized gene accession or lack associated reactions (409), or are mutated in ways that cannot be represented in the metabolic model (76). The majority of LASER genes can be associated with the correct species or genus in Biocyc (71% success), although this metric is biased by the fact that modification of native genes is quite common and are much simpler to match as a result. The matching process adds a total of 963 reactions from Biocyc

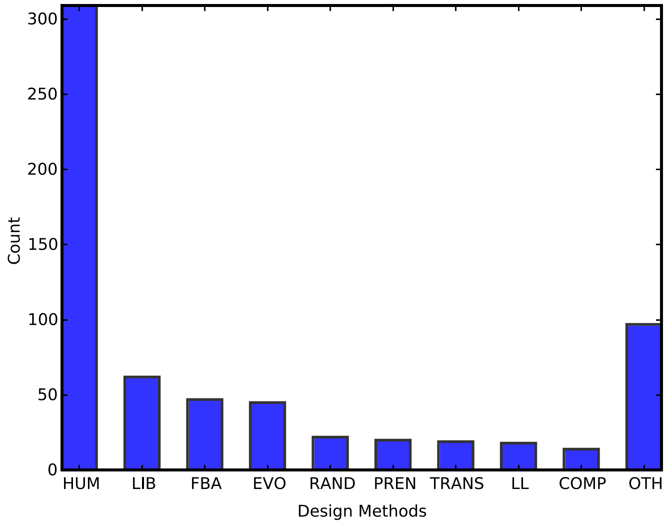

Fig. 5. Distribution of methods used for engineering biocatalysts or improving production. All methods that were employed fewer than 10 times were grouped into the 'Other' (OTH) category. HUM: human designed or inferred, LIB: random or exhaustive library screening, FBA: flux balance analysis, EVO: evolutionary engineering, RAND: random mutagenesis and screening, PREN: protein engineering, TRANS: transcriptomic profiling, LL: liquid–liquid product extraction, COMP: computational (generally elementary mode or non-FBA modeling).

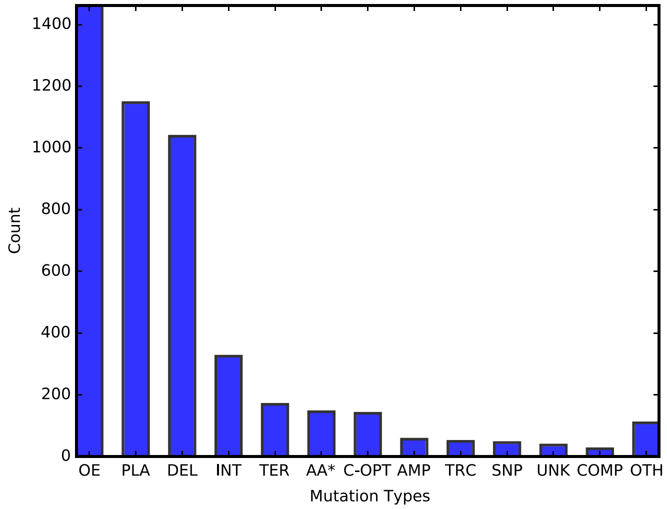

Fig. 6. Distribution of mutations implemented by metabolic engineers. All mutations made less than 20 times are combined into the 'Other' (OTH) category. OE: overexpression, PLA: plasmid, DEL: deletion, INT: genomic integration, TER: gene termination, AA*: amino acid changes, C-OPT: codon optimization, AMP: gene amplification, TRC: gene truncation, SNP: nucleotide changes, UNK: an unspecified mutant allele, COMP: protein compartmentalization.

databases into the COBRA model, with 3960 of the involved metabolites being converted to their COBRA equivalents and 1360 automatically generated from their Biocyc compound information. The regulatory model generation is comparatively more straightforward: 1471 native genes (56%) can be matched and modified in the supplied regulatory networks, with the remainder being genes not present in the network (884 genes) or those with unimplemented mutations that cannot be represented in regulatory network (246 genes).

The sources of these inaccuracies are numerous, but the main cause is that it is often difficult to uniquely pair curated gene names with those existing in external databases. This same obstacle extends to identifying the correct reactions and metabolites to insert into a COBRA model. Error rates in pairing LASER mutation entries to elements in Biocyc and the COBRA models are not unacceptably high, but manual curation is being applied to better link these databases and to improve COBRA-Biocyc metabolite mapping. A similar issue also exists with implementing modifications to regulatory networks, as gene pairing based on inconsistent naming schemes can often be difficult. In both model types, it can also be challenging to implement mutations that are not deletions, repression, or overexpression, such as residue or nucleotide changes in coding sequences due to model simplifications. It may be necessary to examine the stated effect of these modifications (which must be provided during the initial submission in adherence to the LASER schema) and modify the underlying models accordingly. Overall, we expect continual improvements in the accuracy of model generation as LASER is updated.

4. Discussion

The formal LASER design is a key part of the transition from "bespoke" design efforts to more consistent approaches used by other engineering fields (Martin and Ishii, 2002), and would facilitate far more efficient design exchange, evaluation, and archival storage than what is currently possible in the ME field. This need is well known, as Woodruff and colleagues recently highlighted the need to facilitate sharing of validated designs (Woodruff et al., 2013), as others have done on various levels (Keasling, 2008; Shetty et al., 2008). A recent successful effort to formalize metabolic models using the Systems Biology Markup Language (Hucka et al., 2003) and the resulting proliferation of easily read and modified metabolic models demonstrates the potential utility of this approach in the broader metabolic engineering field. Part specification is also becoming increasingly systematic due to the introduction of the related SBOL standard (Galdzicki et al., 2014). LASER designs are easily defined, transferred, and should be easily re-implemented in the laboratory, providing an initial foundation to extend the benefits of standardization to the metabolic engineering field.

The combination of design standardization and automated model generation using LASER enables the development of novel engineering tools. Our efforts to extend LASER are focused on three principal areas: complexity, the difficulty in implementing a given ME design or the feasibility of new ones, forecasting, identifying new targets for research effort in the metabolic engineering field, and finally synthesis, inferring empirical design rules from the database and applying them to develop new strains.

4.1. Complexity

A key question facing metabolic engineers in academia and industry is how to design the simplest strain, usually in terms of the number and type of modifications required to achieve a specified design goal. As detailed above, these judgments are often based on prior experience with a strain, product, or product class as well as assumptions about what genetic modifications will be required to achieve the desired strain performance. Metrics to quantify the complexity of a proposed or existing design are currently lacking, so it is not possible to rank designs according to their difficulty of implementation except using pure empiricism. The LASER corpus can be used to convert conventional wisdom in this area to empirical "laws" of complexity by developing topological metrics that can be correlated to human-perceived complexity. Defining such ME complexity and then using this to predict the feasibility of new projects is an area under active investigation in our lab.

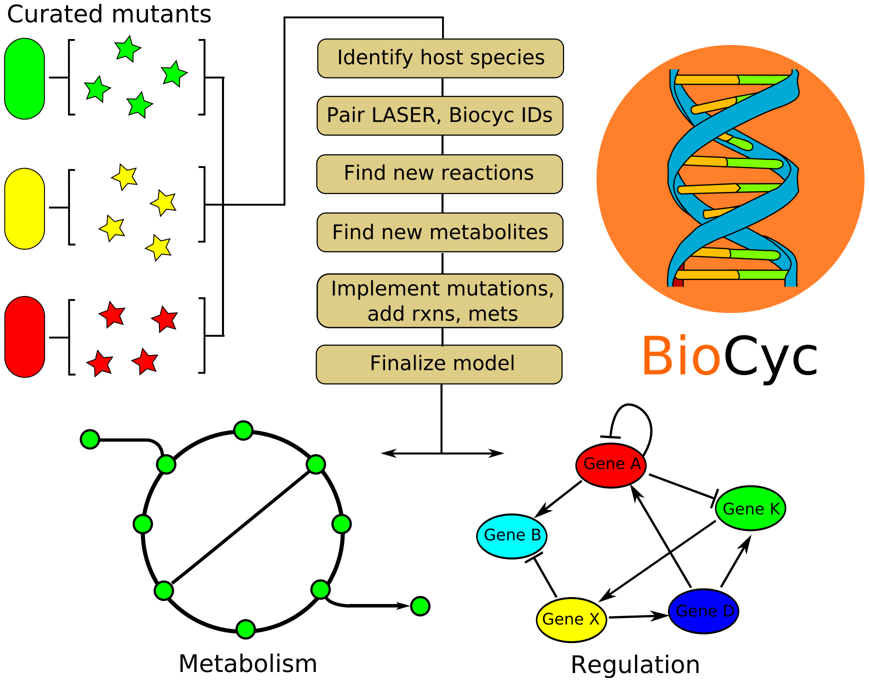

Fig. 7. Schematic representation of the LASER-model conversion process. LASER records are fed into a Python-based pipeline to uniquely identify genes, locate any new reactions and metabolites associated with heterologous genes, and manipulate the corresponding metabolic and regulatory models according to the specified mutation lists associated with each gene.

4.2. Forecasting

The majority of LASER designs are associated with experimental performance data, such as final product titer or yield from the substrate of interest. Combining this data with the generated metabolic models allows users to calculate key metabolic performance parameters (% of theoretical yield achieved, for example) automatically, and then compare their results to the literature without the need for laborious manual curation. It will therefore be possible to infer trends in metabolic engineering subfields in order to identify areas where new technological developments have been brought to bear and those in which progress is comparatively lagging behind. These data can also be used, for example, to calculate the variability in strain performance associated with different design parameters or culture conditions, check consistency with theoretical pathway calculations, and identify key loci associated with improved titers or yields. These quality assurance tools already exist in other fields to aid engineers, and they will likely significantly improve the reproducibility and completeness of published designs once in wider use.

4.3. Synthesis

It is not sufficient to simply analyze past designs; one of the principal goals of LASER is to combine the empirical knowledge we have accumulated as metabolic engineers with current modeling tools to identify new routes for strain improvement. Initially, we will focus on selecting common chassis strains based on the LASER models, followed by combining metabolic modeling and previous empirical results to improve yields. Subsequent analyses will aim to develop data-driven tools that can propose designs from a user-provided product, medium, and host strain that will represent the best possible synthesis of experimental and modeling-guided design. A simple example of this approach is to analyze LASER and extract the frequency at which specific genes are modified, and their association probability as shown in Fig. 8. This exercise, while purely statistical, generates results pointing to the common need to balance redox metabolism and hints at reasonably standard designs involving a limited set of permutations of *ldhA*, *pta*, *frdABCD*, *adhE*, and *ackA* deletions. The effect of individual modifications can also be imputed by comparing distances between designs (e.g. number of genetic modifications separating them) to see how product yield and titer were effected, all else being equal. This type of mutation effect analysis will be explored in future work.

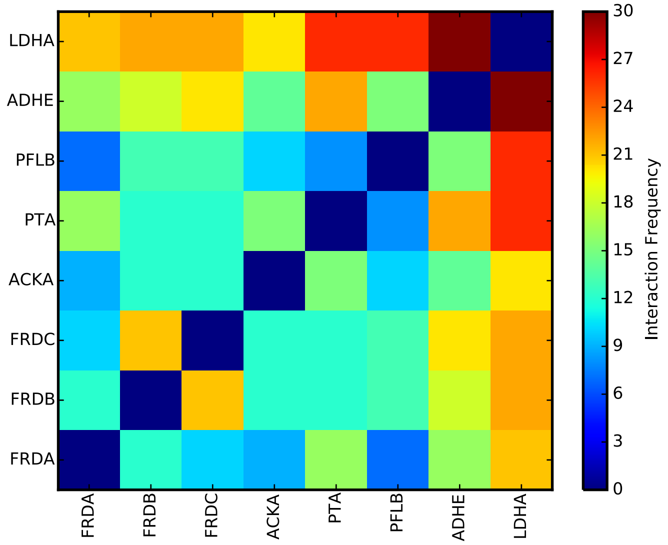

Fig. 8. Pairwise association between commonly mutated genes. Many *E. coli* designs rely on manipulation of redox balancing to improve production of particular compounds (organic acids, ethanol, n-butanol, and others) via growth essentially or reduced flux diversion, with these genes representing the most common routes for acetate, ethanol, pyruvate, and lactate dissimilation and hence, redox manipulation.

5. Conclusions

In LASER, we have developed a database of curated metabolic engineering designs, alongside a definition for what constitutes a metabolic engineering design. Over 300 papers and 417 designs, representing the known bibliome of papers containing genetically defined yeast and *E. coli* designs, have been curated thus far, providing an immense resource of metabolic engineering knowledge. However, this collection is certainly not complete, and we welcome metabolic engineers and synthetic biologists to deposit their designs into LASER from past research and current projects. A public interface for the database has been setup at laser.colorado.edu to facilitate this effort and provide a public platform for design storage and analysis. We anticipate that the LASER database and its associated tools will enable new types of strain design, analysis, and improved project management hereto thought to be impractical.

Acknowledgments

We thank Emily F. Freed and Marcelo Bassalo with their assistance with curating metabolite and reaction mappings for the matching pipeline. This project was financially supported by the Department of Energy Genome Science Program (award #: DE-SCOOO8812).

Appendix A. Supplementary data

Supplementary data associated with this article can be found in the online version at http://dx.doi.org/10.1016/j.meteno.2015.06.003.

References

Alper, H., Miyaoku, K., Stephanopoulos, G., 2005. Construction of lycopene-overproducing *E. coli* strains by combining systematic and combinatorial gene knockout targets. Nat. Biotechnol. 23 (5), 612–616.

Andrianantoandro, E., Basu, S., Karig, D.K., Weiss, R., 2006. Synthetic biology: new engineering rules for an emerging discipline. Mol. Syst. Biol. 2 (1).

Antoniewicz, M.R., 2015. Methods and advances in metabolic flux analysis: a mini-review. J. Ind. Microbiol. Biotechnol., pp. 1–9.

Armstrong, J., 2007. Programming Erlang: Software for a Concurrent World. Pragmatic Bookshelf, New York, USA.

Bailey, J.E., 1991. Toward a science of metabolic engineering. Science 252 (5013), 1668–1675.

Bilitchenko, L., Liu, A., Cheung, S., Weeding, E., Xia, B., Leguia, M., Anderson, J.C., Densmore, D., 2011. Eugene—a domain specific language for specifying and constraining synthetic biological parts and systems. PloS ONE 6 (4), e18882.

Bush, M., 1990. Improving software quality: the use of formal inspections at the JPL. In: Proceedings of the 12th International Conference on Software Engineering. IEEE Computer Society Press, Nice, France, pp. 196-199.

Cambria, E., White, B., 2014. Jumping NLP curves: a review of natural language processing research. IEEE Comput. Intell. Mag. 9 (2), 48–57.

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

Caspi, R., Altman, T., Billington, R., Dreher, K., Foerster, H., Fulcher, C.A., Holland, T.A., Keseler, I.M., Kothari, A., Kubo, A., Krummenacker, M., Latendresse, M., Mueller, L.A., Ong, Q., Paley, S., Subhraveti, P., Weaver, D.S., Weerasinghe, D., Zhang, P., Karp, P.D., 2014. The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of Pathway/Genome Databases. Nucl. Acids Res. 42 (D1), D459–D471.

Choi, Y.-K., 2004. In: Principles of Applied Civil Engineering Design. ASCE, New York, USA .

Costanzo, M.C., Engel, S.R., Wong, E.D., Lloyd, P., Karra, K., Chan, E.T., Weng, S., Paskov, K.M., Roe, G.R., Binkley, G., Hitz, B.C., Cherry, J.M., 2014. Saccharomyces genome database provides new regulation data. Nucl. Acids Res. 42 (D1), D717–D725.

Doudna, J.A., Charpentier, E., 2014. The new frontier of genome engineering with CRISPR-Cas9. Science 346 (6213), 1258096.

Easley, D., Lopez de Prado, M., O'Hara, M., 2011. The microstructure of the 'Flash Crash': flow toxicity, liquidity crashes and the probability of informed trading. J. Portf. Manag. 37 (2), 118–128.

Ebrahim, A., Lerman, J.A., Palsson, B.O., Hyduke, D.R., 2013. COBRApy: constraints-based reconstruction and analysis for Python. BMC Syst. Biol. 7 (1), 74.

Endy, D., 2005. Foundations for engineering biology. Nature 438 (7067), 449–453.

Fisher, A.K., Freedman, B.G., Bevan, D.R., Senger, R.S., 2014. A review of metabolic and enzymatic engineering strategies for designing and optimizing performance of microbial cell factories. Comput. Struct. Biotechnol. J. 11 (18), 91–99.

Frushicheva, M.P., Mills, M.J., Schopf, P., Singh, M.K., Prasad, R.B., Warshel, A., 2014. Computer aided enzyme design and catalytic concepts. Curr. Opin. Chem. Biol. 21, 56–62.

Galdzicki, M., Clancy, K.P., Oberortner, E., Pocock, M., Quinn, J.Y., Rodriguez, C.A., Roehner, N., Wilson, M.L., Adam, L., Anderson, J.C., Bartley, B., Beal, J., Chandran, D., Chen, J., Densmore, D., Endy, D., Grunberg, R., Hallinan, J., Hillson, N., Johnson, J., Kuchinsky, A., Lux, M., Misirli, G., Peccoud, J., Plahar, H., Sirin, E., Stan, G., Villalobos, A., Wipat, A., Gennari, J., Myers, C., Sauro, H., 2014. The Synthetic Biology Open Language (SBOL) provides a community standard for communicating designs in synthetic biology. Nat. Biotechnol. 32 (6), 545–550.

Gosling, J., McGilton, H., 1995. The Java Language Environment. Sun Microsystems Computer Company, Mountain View, CA.

Green, D., Perry, R., 2007. Perry's Chemical Engineers' Handbook, Eighth Edition McGraw Hill Professional, McGraw-Hill Education, New York, USA.

Gronenberg, L.S., Marcheschi, R.J., Liao, J.C., 2013. Next generation biofuel engineering in prokaryotes. Curr. Opin. Chem. Biol. 17 (3), 462–471.

Ham, T.S., Dmytriv, Z., Plahar, H., Chen, J., Hillson, N.J., Keasling, J.D., 2012. Design, implementation and practice of JBEI-ICE: an open source biological part registry platform and tools. Nucl. Acids Res. 40 (18), e141.

Hayashi, M., Tabata, K., 2013. Metabolic engineering for l-glutamine overproduction by using DNA gyrase mutations in *Escherichia coli*. Appl. Environ. Microbiol. 79 (9), 3033–3039.

Heavner, B.D., Smallbone, K., Barker, B., Mendes, P., Walker, L.P., 2012. Yeast 5—an expanded reconstruction of the saccharomyces cerevisiae metabolic network. BMC Syst. Biol. 6 (1), 55.

Henry, C.S., DeJongh, M., Best, A.A., Frybarger, P.M., Linsay, B., Stevens, R.L., 2010. High-throughput generation, optimization and analysis of genome-scale metabolic models. Nat. Biotechnol. 28 (9), 977–982.

Hesselman, M.C., Koehorst, J.J., Slijkhuis, T., Odoni, D.I., Hugenholtz, F., van Passel, M.W., 2012. The constructor: a web application optimizing cloning strategies based on modules from the registry of standard biological parts. J. Biol. Eng. 6 (1), 14.

Highsmith, J., Cockburn, A., 2001. Agile software development: the business of innovation. Computer 34 (9), 120–127.

Hucka, M., Finney, A., Sauro, H.M., Bolouri, H., Doyle, J.C., Kitano, H., the rest of the SBML Forum:Arkin, A.P., Bornstein, B.J., Bray, D., Cornish-Bowden, A., Cuellar, A. A., Dronov, S., Gilles, E.D., Ginkel, M., Gor, V., Goryanin, I.I., Hedley, W.J., Hodgman, T.C., Hofmeyr, J.-H., Hunter, P.J., Juty, N.S., Kasberger, J.L., Kremling, A., Kummer, U., Le Novère, N., Loew, L.M., Lucio, D., Mendes, P., Minch, E., Mjolsness, E.D., Nakayama, Y., Nelson, M.R., Nielsen, P.F., Sakurada, T., Schaff, J. C., Shapiro, B.E., Shimizu, T.S., Spence, H.D., Stelling, J., Takahashi, K., Tomita, M., Wagner, J., Wang, J., 2003. The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. Bioinformatics 19 (4), 524–531.

Hunter, J.D., 2007. Matplotlib: a 2D graphics environment. Comput. Sci. Eng. 9 (3), 90–95.

Jones, J.A., Toparlak, Ö.D., Koffas, M.A., 2015. Metabolic pathway balancing and its role in the production of biofuels and chemicals. Curr. Opin. Biotechnol. 33, 52–59.

Keasling, J.D., 2008. Synthetic biology for synthetic chemistry. ACS Chem. Biol. 3 (1), 64–76.

Keasling, J.D., 2010. Manufacturing molecules through metabolic engineering. Science 330 (6009), 1355–1358.

Khoury, G.A., Smadbeck, J., Kieslich, C.A., Floudas, C.A., 2014. Protein folding and *de novo* protein design for biotechnological applications. Trends Biotechnol. 32 (2), 99–109.

Leveson, N.G., Turner, C.S., 1993. An investigation of the Therac-25 accidents. Computer 26 (7), 18–41.

Lynch, M.D., Warnecke, T., Gill, R.T., 2007. SCALEs: multiscale analysis of library enrichment. Nat. Methods 4 (1), 87–93.

Marienhagen, J., Bott, M., 2013. Metabolic engineering of microorganisms for the synthesis of plant natural products. J. Biotechnol. 163 (2), 166–178.

Martin, M.V., Ishii, K., 2002. Design for variety: developing standardized and modularized product platform architectures. Res. Eng. Des. 13 (4), 213–235.

Martin, V.J., Pitera, D.J., Withers, S.T., Newman, J.D., Keasling, J.D., 2003. Engineering a mevalonate pathway in *Escherichia coli* for production of terpenoids. Nat. Biotechnol. 21 (7), 796–802.

Mills, H.D., Linger, R.C., 1991. Cleanroom Software Engineering: Developing Software under Statistical Quality Control. Wiley Online Library, New York, USA.

Nielsen, J., 2013. Production of biopharmaceutical proteins by yeast: advances through metabolic engineering. Bioengineered 4 (4), 207–211.

Orth, J.D., Conrad, T.M., Na, J., Lerman, J.A., Nam, H., Feist, A.M., Palsson, B.Ø., 2011. A comprehensive genome-scale reconstruction of escherichia coli metabolism—2011. Mol. Syst. Biol. 7 (1).

Pál, C., Papp, B., Pósfai, G., 2014. The dawn of evolutionary genome engineering. Nat. Rev. Genet. 15 (7), 504–512.

Salgado, H., Peralta-Gil, M., Gama-Castro, S., Santos-Zavaleta, A., Muñiz-Rascado, L., García-Sotelo, J.S., Weiss, V., Solano-Lira, H., Martínez-Flores, I., Medina-Rivera, A., Salgado-Osorio, G., Alquicira-Hernández, S., Alquicira-Hernández, K., López-Fuentes, A., Porrón-Sotelo, L., Huerta, A.M., Bonavides-Martínez, C., Balderas-Martínez, Y.I., Pannier, L., Olvera, M., Labastida, A., Jiménez-Jacinto, V., Vega-Alvarado, L., del Moral-Chávez, V., Hernández-Alvarez, A., Morett, E., Collado-Vides, J., 2013. RegulonDB v8.0: omics data sets evolutionary, conservation, regulatory phrases, cross-validated gold standards and more. Nucl. Acids Res. 41 (D1), D203–D213.

Schellenberger, J., Que, R., Fleming, R.M., Thiele, I., Orth, J.D., Feist, A.M., Zielinski, D. C., Bordbar, A., Lewis, N.E., Rahmanian, S., Kang, J., Hyduke, D., Palsson, B., 2011. Quantitative prediction of cellular metabolism with constraint-based models:

the cobra toolbox v2. 0. Nat. Protoc. 6 (9), 1290–1307.

Schomburg, I., Chang, A., Placzek, S., Söhngen, C., Rother, M., Lang, M., Munaretto, C., Ulas, S., Stelzer, M., Grote, A., Scheer, M., Schomburg, D., 2013. BRENDA in 2013: integrated reactions, kinetic data, enzyme function data, improved disease classification: new options and contents in BRENDA. Nucl. Acids Res. 41 (D1), D764–D772.

Shaw, M., Garlan, D., 1996. Software Architecture: Perspectives on an Emerging Discipline, vol. 1. Prentice Hall Englewood Cliffs, NJ, USA.

Shetty, R.P., Endy, D., Knight Jr, T.F., 2008. Engineering BioBrick vectors from BioBrick parts. J. Biol. Eng. 2 (1), 1–12.

Shigley, J.E., 2011. Shigley's Mechanical Engineering Design. Tata McGraw-Hill Education, New York, USA.

Sinnott, R.K., 2009. Chemical Engineering Design: SI Edition. Elsevier, Oxford, UK.

Teixeira, M.C., Monteiro, P.T., Guerreiro, J.F., Gonçalves, J.P., Mira, N.P., dos Santos, S. C., Cabrito, T.R., Palma, M., Costa, C., Francisco, A.P., Madeira, S.C., Oliveira, A.L., Freitas, A.T., Sá-Correia, I., 2014. The YEASTRACT database: an upgraded information system for the analysis of gene and genomic transcription regulation in *Saccharomyces cerevisiae*. Nucl. Acids Res. 42 (D1), D161–D166.

Towler, G.P., Sinnott, R.K., 2013. Chemical Engineering Design: Principles, Practice, and Economics of Plant and Process Design. Elsevier, New York, USA.

Trinh, C.T., Wlaschin, A., Srienc, F., 2009. Elementary mode analysis: a useful metabolic pathway analysis tool for characterizing cellular metabolism. Appl. Microbiol. Biotechnol. 81 (5), 813–826.

Warner, J.R., Reeder, P.J., Karimpour-Fard, A., Woodruff, L.B., Gill, R.T., 2010. Rapid profiling of a microbial genome using mixtures of barcoded oligonucleotides. Nat. Biotechnol. 28 (8), 856–862.

Winkler, J., Kao, K.C., 2012. Harnessing recombination to speed adaptive evolution in *Escherichia coli*. Metab. Eng. 14 (5), 487–495.

Woodruff, L., May, B.L., Warner, J.R., Gill, R.T., 2013. Towards a metabolic engineering strain "commons": an *Escherichia coli* platform strain for ethanol production. Biotechnol. Bioeng. 110 (5), 1520–1526.

Woolston, B.M., Edgar, S., Stephanopoulos, G., 2013. Metabolic engineering: past and future. Annu. Rev. Chem. Biomol. Eng. 4, 259–288.

Zeitoun, R.I., Garst, A.D., Degen, G.D., Pines, G., Mansell, T.J., Glebes, T.Y., Boyle, N. R., Gill, R.T., 2015. Multiplexed tracking of combinatorial genomic mutations in engineered cell populations. Nat. Biotechnol., http://dx.doi.org/10.1038/nbt.3177.