ELSEVIER

Computational tools for metabolic engineering

Wilbert B. Copeland [a,*], Bryan A. Bartley [a], Deepak Chandran [a], Michal Galdzicki [b],
Kyung H. Kim [a], Sean C. Sleight [a], Costas D. Maranas [c], Herbert M. Sauro [a]

[a] Department of Bioengineering, University of Washington, William H. Foege Building, Box 355061, 3720 15th Ave NE, Seattle, WA 98195-5061, USA
[b] Biomedical and Health Informatics, University of Washington, Box 357240, 1959 NE Pacific Street, HSB I-264, Seattle, WA 98195-7240, USA
[c] Department of Chemical Engineering, Penn State, 112A Fenske Laboratory, University Park, PA 16802, USA

ARTICLE INFO

ABSTRACT

A great variety of software applications are now employed in the metabolic engineering field. These applications have been created to support a wide range of experimental and analysis techniques. Computational tools are utilized throughout the metabolic engineering workflow to extract and interpret relevant information from large data sets, to present complex models in a more manageable form, and to propose efficient network design strategies. In this review, we present a number of tools that can assist in modifying and understanding cellular metabolic networks. The review covers seven areas of relevance to metabolic engineers. These include metabolic reconstruction efforts, network visualization, nucleic acid and protein engineering, metabolic flux analysis, pathway prospecting, post-structural network analysis and culture optimization. The list of available tools is extensive and we can only highlight a small, representative portion of the tools from each area.

© 2012 Elsevier Inc. All rights reserved.

1. Introduction

There is a long tradition of using software in metabolic research, starting with analog computers to model single enzymes (Chance, 1943; Chance et al., 1952), digital computers to model the dynamics of metabolic pathways (Chance et al., 1960), and in more recent decades, the use of bioinformatics tools to search and query large metabolic databases (Karp and Caspi, 2011). Advancements in metabolic engineering have led to an expansive collection of computational tools that assist in the rational engineering of cellular metabolism, and the utility of these tools becomes increasingly apparent when managing, analyzing, and visualizing large metabolic pathways. In this review, we will discuss common tasks encountered by metabolic engineers and we will highlight computational tools that support researchers attempting to complete those tasks (Fig. 1). One growing area of computation which is finding increasing use in metabolic engineering is protein and enzyme design. In this review we will not be considering this particular topic but refer the reader to the work by Kortemme and Baker (2004).

2. Reconstructing metabolic networks

Metabolic network reconstructions (Henry et al., 2010; Thiele and Palsson, 2010) describe the identity and function of constituents involved in an organism's metabolism by linking metabolic and genomic data. The scale of network reconstructions may range from individual pathways to whole-genomes. The size of a network and the quality of annotation information are significant factors that affect the thoroughness of a reconstruction. Analyzing and annotating genomic sequences, storing and retrieving metabolic network information, and representing network data are key tasks associated with metabolic network reconstruction, and will be discussed below. Whether individuals choose to manually reconstruct a metabolic network or to draw from the work of others, network reconstruction is the essential first step in any metabolic engineering application.

2.1. Building network reconstructions

One route to uncover details of a novel metabolic pathway involves comparing the unknown network with already well-characterized networks. The utility of comparative analysis strategies in quickly reconstructing a network has been highlighted in situations such as understanding C_3 and C_4 metabolism in *Methylobacterium extorquens* AM1 (Van Dien et al., 2003) or expanding the description of a genome-scale metabolic network in *Lactococcus lactis* IL1403 (Notebaart et al., 2006). This strategy typically begins with researchers employing programs or databases that implement **BLAST** (Altschul et al., 1990; Johnson et al., 2008), an open-source local sequence alignment method that searches for statistically significant matches between a query sequence and a collection of sequences in a database. This strategy

* Correspondence author. Fax: +206 543 4791.
E-mail address: wcopelan@uw.edu (W.B. Copeland).
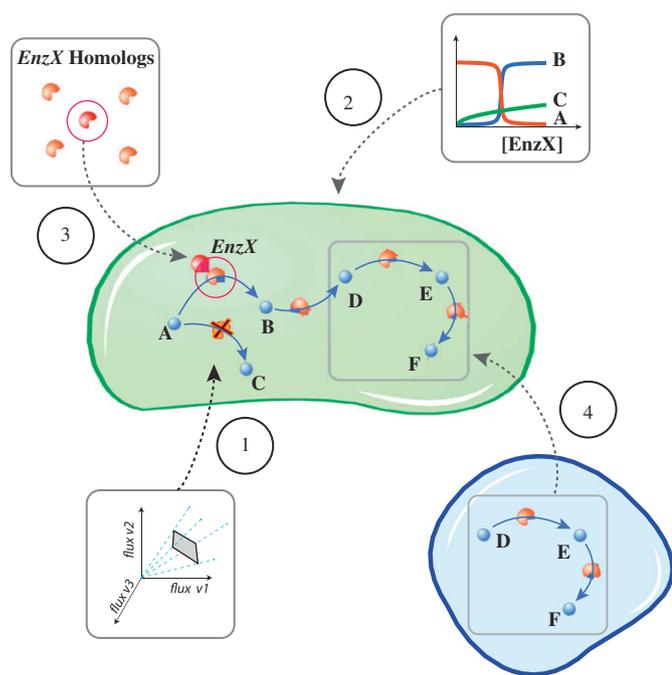
Fig. 1. To optimize the production of molecular compounds, a variety of metabolic engineering strategies can be augmented by computational tools. (1) Targeted knockouts of enzymes in non-essential pathways are assisted by constraint-based approaches that maximize cell growth and product yields. (2) Dynamic simulation strategies suggest regions in a metabolic pathway that are particularly amenable to modification. (3) Network reconstruction tools produce collections of metabolic models that can be screened for heterologous enzymes, and biological molecule manipulation tools ensure efficient expression of promising enzyme candidates within engineered cells. (4) Pathway prospecting tools aid researchers looking to integrate complex reaction pathways into non-native hosts.

requires the metabolic engineer to have access to nucleotide or protein sequence data, and assumes that homologous sequences in the unknown and known networks are likely to encode proteins of similar function. Manual comparative analysis can be performed using the NCBI's suite of web-based BLAST implementations (Johnson et al., 2008), or users may opt to use various software tools such as **KAAS** (Moriya et al., 2007) or **PathoLogic** (Karp et al., 2002) to automate a reconstruction. These automated tools use BLAST and other comparative analysis algorithms in conjunction with information contained in specific databases such as **KEGG** (Kanehisa et al., 2006) or **BioCyc** (Caspi et al., 2008, 2010) to produce a list of predicted orthologous genes.

Comparative analysis methods, by themselves, often yield inadequate network reconstructions. To make accurate predictions about a metabolic network, researchers often refine the reconstruction using experimental approaches. Readers are referred to an excellent review by Thiele and Palsson on the enrichment procedure for generating high-quality network reconstructions (Thiele and Palsson, 2010).

Currently, the speed at which metabolic network reconstructions are generated is eclipsed by the speed at which genomes are sequenced. To keep pace, automated computational strategies are needed. The **SEED** framework (DeJongh et al., 2007) seeks to enable high-throughput generation of genome-scale metabolic models that are complete and coherent enough for systems level analysis. **Model SEED** (Henry et al., 2010) integrates genome annotations, gene–protein-reaction associations, biomass reactions, and thermodynamic analysis of reversibility to assemble a reaction network's topology. To accomplish this, Model SEED draws on existing databases as well as tools contained on its server. During automated construction, Model SEED identifies structural inconsistencies in the reconstructed model and

identifies the minimal set of reactions required to amend those discrepancies using data obtained from various databases. Interestingly, Model SEED uses the information gathered during gap analysis to simultaneously and iteratively enrich both genome annotation data and the network model. As an information resource, Model SEED (Henry et al., 2010) provides drafted model reconstructions for download in a variety of formats including **SBML** (Hucka et al., 2003).

As discussed, the quality of a network reconstruction is greatly affected by annotated sequences. Researchers and database tools use annotation data to associate nucleotide and protein sequences with their respective functions during metabolic network reconstructions. Ontologies establish a way to annotate metabolic pathway components so that computational tools can quickly and accurately interpret the relationship between biological entities. Computational tools can leverage ontological annotations to increase the efficiency of information analysis and retrieval tasks such as mapping the functional description between orthologous enzymes. The widely adopted **Gene Ontology** (**GO**; Ashburner et al., 2000) is used to annotate gene and protein sequences according to the properties of the protein product. GO captures protein properties on three levels: (1) the cellular component annotation describes the molecule's location or family, (2) the molecular function annotation describes the molecule's specific role, and (3) the biological process annotation describes the broader cellular process in which a particular molecule participates. In addition to using GO to describe gene product properties, genomic sequence annotation can adhere to a set of standardized terms and relationships set forth by the **Sequence Ontology** (**SO**; Eilbeck et al., 2005). SO establishes a controlled vocabulary for describing nucleotide and amino acid sequence features and attributes. Use of SO helps reduce ambiguity when annotating genomic sequences and assists software tools in inferring the relationship between sequences.

2.2. Storing and retrieving network information

Producing genome-scale metabolic network models is a significant research effort (Durot et al., 2009), so communal information resources have been developed to make existing network reconstruction models readily available to metabolic engineers. Two highly-referenced and extensive collections of metabolic networks for a large number of organisms are provided by **KEGG PATHWAY** (Ogata et al., 1999; Kanehisa and Goto, 2000; Kanehisa et al., 2006) and **MetaCyc** (Caspi et al., 2008, 2010). KEGG Pathway and MetaCyc both display networks as a reaction map that describes interactions between enzymes and substrates. KEGG Pathway contains manually drawn networks that represent a reference metabolic pathway rather than a network unique to a particular organism. In contrast, MetaCyc automatically generates organism-specific metabolic network diagrams and provides a summary of relevant literature references for proposed reactions. Metabolic engineers can browse or review a network's reactions and metabolites on the web prior to use in downstream analysis. Both collections are curated from experimental literature and linked to gene databases by EC numbers. Individuals interested in metagenomic prospecting should find each tool has sufficient methods to accelerate cross-species comparisons of enzymes involved in a general pathway class.

Although once an entirely free resource, only KEGG Pathway's web interface remains free, and access to download the database requires a paid subscription. Pathway models from KEGG can be used for genome reconstruction and subsequent analysis by exporting in KGML. For most software tools to use KGML data, it must first be converted into a standard file format using a tool such as **KEGG translator** (Wrzodek et al., 2011). In addition to its

web interface, enzyme and reaction information in MetaCyc can be queried, compared, and visualized using the cross-platform program, **Pathway Tools** (Karp et al., 2010).

The difficulty in merging network information from many repositories stems from discordant pathway representations among various sources. These discrepancies lead to duplications and erroneous predictions that can be mitigated, in part, by using a consistent data representation scheme. **MetRxn** (Kumar et al., 2011) and the **Biochemical, Genetic, and Genomic** knowledgebase (**BiGG**; Schellenberger et al., 2010) are two database tools that seek to integrate genome-scale metabolic network reconstructions under a standard network description scheme. MetRxn uses an automated process to integrate knowledge from eight databases and forty-four metabolic models, whereas BiGG contains seven manually curated reconstructions. Major steps in MetRxn's automation strategy involve comparing atom bond connectivity of metabolites and balancing elements and charges for reactions. Models in BiGG are mass and charge balanced, and they contain pathway information such as compartment localization, EC number, reversibility, and the relationship between genes, proteins, and reactions (Schellenberger et al., 2010). Their creators maintain that the consistency, quality, and abundance of information contained in either of these databases should greatly enhance the process by which metabolic engineers browse and analyze large-scale metabolic networks.

2.3. Knowledge and information representation standards

To support the metabolic engineering pipeline, information and data should be easily propagated among computational tools and databases. Information technology standards provide this capability by specifying the syntax and semantics in a standard form (Sauro, 2008). The **Systems Biology Markup Language** (**SBML**; Hucka et al., 2003) is a common format used to represent metabolic pathway models. Since its development in 2000, it has experienced widespread use, and as of this review 222 tools support SBML. The **Systems Biology Ontology** (**SBO**; Le Novère, 2011) is used to specify the meaning of parameters or equations contained in kinetic models of a biochemical pathway. The **Biological Pathway Exchange Language** (**BioPAX**; Demir et al., 2010) defines the semantics for pathway representation, which allows databases and software tools to exchange information regarding the qualitative functional relationships of networks.

3. Visualizing metabolic networks

Network visualization allows researchers to leverage the power of human intuition to identify patterns that are undetectable by computer algorithms. Network visualization serves as a bridge that links metabolic network reconstructions to downstream analysis and design methods. Most of the databases discussed in the previous section implement some form of visualization to help researchers browse network reconstruction content, and the visualization tools described in this section typically display metabolic network data derived from databases. With respect to biochemical systems, including metabolic networks, there are two types of network visualization schemes that are supported by software tools: interaction networks and reaction networks. Interaction networks provide researchers with a map of all the molecular interactions in the system of interest, whereas reaction networks depict the stoichiometry in a biochemical system.

Metabolic engineers typically view networks as enzymatic processes where one metabolite is converted to another via a catalyst. Visualizing metabolic pathways using a reaction network

scheme as opposed to using an interaction network scheme aids researchers in identifying properties of the system such as positive and negative feedback, which are key properties that cannot be inferred easily from interaction networks. Nevertheless, **Cytoscape** (Shannon et al., 2003; Kohl et al., 2011), **GraphViz** (Gansner and North, 2000), and **Systrip** (Auber, 2003) are popular interaction network visualization software tools. These tools employ algorithms that automatically lay out large interaction networks. Cytoscape has extensive plugin support, whereas GraphViz and Systrip are easily incorporated into other software tools. **CytoSEED** (DeJongh et al., 2011) is one particularly useful Cytoscape plugin, as it allows users to visualize models created in Model SEED.

3.1. Reaction networks

Metabolic pathways are commonly depicted using reaction network diagrams rather than interaction network diagrams because the stoichiometry of enzyme catalyzed reactions is not usually captured by interaction networks. Further, it is often the case that reaction network diagrams are used in conjunction with mathematical models; therefore dynamical properties of a network, such as fluxes, metabolite concentrations, and feedback regulation, can be visualized and simulated. All of the reaction network visualization tools discussed below display reactions using either Bézier curves or line segments, and apply arrowheads to these segments to distinguish reactants from products. Some of the discussed emphasize reactions by drawing regulatory arcs from enzymes to the reactions they catalyze. Finally, researchers using most of the tools opt to store and retrieve reaction network information based on the SBML file format (Hucka et al., 2003).

GLAMM (Bates et al., 2011) and **Vanted** (Klukas and Schreiber, 2010), are two reaction network visualization tools that incorporate experimental data. Vanted is an open-source and cross-platform package written in Java, and GLAMM is a web-based tool. These tools have particular utility for researchers interested in observing gene regulation within the context of metabolic pathways under a range of environmental conditions. Both tools allow users to overlay metabolic network reconstructions with experimental data that pertains to specific reactions or regions of a pathway. GLAMM draws on genomic data found on **MicrobesOnline** (Dehal et al., 2010), whereas Vanted integrates data from KEGG, GO, and imported files of standard formats such as SBML.

Another class of network visualization tools is tailored more towards mathematical analysis of reaction networks. Their diagrams correspond to an underlying mathematical model, and this linkage often makes their diagrams more restrictive than other types of visualization software. **Arcadia** (Villéger et al., 2010) is an open-source, cross-platform application primarily used to view biochemical systems and their associated parameters, as specified by an SBML file. Although **CellNetAnalyzer** (Klamt et al., 2007) is primarily a constraint-based flux analysis tool, it also excels in network visualization. One notable utility of this tool stems from its ability to display pathway fluxes in addition to reactions and metabolites. **Omix** (Droste et al., 2011) is a rich network visualization tool that is particularly good for visualizing fluxes, and ^{13}C data. In addition, Omix can display regulatory arcs, visualize metabolome data, metabolic flux analysis data, relate metabolic responses to transcriptome data and visualize networks at different abstraction levels. Finally, Omix provides a powerful scripting language interface to interact with the visual representation. Although not open source, the application is freely available under a non-commercial license.

Researchers content with simply viewing a metabolic network should find Arcadia sufficient; however, those looking for computational tools with drawing capabilities to build or modify

metabolic pathway diagrams should consider **JDesigner** (Sauro et al., 2004; Bergmann and Sauro, 2006), **CellDesigner** (Funahashi et al., 2003; 2008), or **Cell Illustrator** (Nagasaki et al., 2010). The capabilities of these tools are similar enough that researchers may favor one tool based on subjective preferences; however, there are minor differences in the way these tools implement visualization tasks. JDesigner takes a minimal approach to representing networks by using one notation for chemical species, which can be decorated with visual cues, and one for reactions, whereas Cell Illustrator and Cell Designer contain multiple types of nodes that represent classes of molecules, such as proteins or DNA, to assist users upon initial visual inspection. JDesigner uses multi-Bézier arcs to render reactions and regulatory interactions in a more visually appealing manner than the line segment diagrams displayed using the other two tools. In addition to viewing and drawing, all three tools possess the ability to simulate networks and estimate parameters based on provided information. Both JDesigner and CellDesigner can import and export models in the form of SBML.

When expressing multiple heterologous enzymes, it may be necessary to visualize the structure of gene networks (Yoon et al., 2007). **TinkerCell** (Chandran et al., 2009, 2010) can be used to view a range of biochemical models, including gene regulation, protein signaling, or metabolic reaction models. TinkerCell uses a customizable set of symbols to distinguish between different classes of molecules and employs different reaction arrowheads to qualify various biochemical processes. In addition to displaying networks, TinkerCell has drawing capabilities to represent DNA segments or entire plasmids which are useful when designing synthetic gene networks; and this ability will be discussed in further detail later. **BioTapestry** (Longabaugh et al., 2009) focuses on representing the interactions at cis-regulatory elements of a gene. This approach allows individuals to view intricacies such as a single transcription factor binding at multiple sites on a gene regulatory module. Interestingly, BioTapestry has the ability to represent networks within their spatial and temporal contexts, and this feature allows for observation of emergent, higher-order motifs.

Establishing standard symbols for visualizing biochemical networks allows for easier access to and interpretation of information displayed in network diagrams. **Systems Biology Graphical Notation** (SBGN; Le Novère et al., 2009) is a proposed standard for representing biochemical network diagrams. Likewise, the **Synthetic Biology Open Language** (SBOL; www.sbol standard.org) has proposed a visual representation to describe DNA components in a gene network diagram, and can be used to build gene networks using parts described in a knowledgebase (Galdzicki et al., 2011).

4. Manipulating biological molecules

Ultimately, implementing changes in a metabolic network requires direct manipulation of biological molecules. Synthesizing nucleic acids and proteins, constructing expression cassettes, and mutating nucleic acid sequences are common metabolic engineering tasks that involve physical modification of biological molecules. A range of experimental methods exist to accomplish these tasks, and software packages have been developed to ensure that those methods are performed under optimal conditions. Computational tools of this type are deployed by researchers to minimize repeated rounds of molecular construction due to off-target interactions, suboptimal enzyme activity, or human errors in sequence design. These tools increase efficiency when designing nucleotides or proteins by linking sequence information with experimentally observed structural or functional states within the context of a host organism.

4.1. Constructing expression cassettes

Cellular production of exogenous enzymes is enabled by the incorporation of protein-coding DNA sequences into the genome or onto a plasmid. For sufficiently complex expression cassettes, researchers may wish to proceed with construction at various levels of abstraction. Functional-level representations of DNA components are useful when viewing cassette architecture or visualizing interactions between gene products expressed on a cassette. Sequence-level representations of DNA components are used to guide molecular construction.

Consider, as an example, a study which engineered lycopene biosynthesis in *E. coli* by expressing carotenoid pathway enzymes as well as enzymes that increased production of carotenoid precursors (Yoon et al., 2007). **TinkerCell** (Chandran et al., 2010) and **GenoCAD** (Czar et al., 2009) are two computational tools that simplify the process of incorporating enzymes into a functional circuit. The primary purpose of either tool is to produce a diagram that describes a gene construct's architecture by using a set of symbols that represent DNA components such as promoters, coding sequences, or terminators. Although both tools choose not to present the DNA sequence in their diagrams, all components can still store sequence information. TinkerCell allows users to assign parameter values to gene network components for simulation. GenoCAD allows researchers to search for DNA components from a catalog and assemble them to design a genetic construct. GenoCAD establishes a grammar to describe components, and uses this grammar to verify the validity of a particular design (Cai et al., 2010).

When devising a strategy to assemble the carotenoid genes into a plasmid, or when verifying the plasmid's sequence following construction, working at the sequence level would be more appropriate. **Geneious** (Drummond et al., 2011) is a sequence editor with powerful DNA visualization, organization, assembly, and alignment features. Geneious users can annotate sequences to highlight features such as promoter sites, origins of replication, or restriction sites. This information is necessary when, for example, a researcher needs to ensure restriction sites are not located within a coding sequence or when manually designing primers to amplify a particular vector region (Sleight et al., 2010). Geneious integrates a host of tools to facilitate cloning and subcloning methods, including Gateway (Karimi et al., 2007), restriction–ligation (Cohen, 1973; Shetty et al., 2008), and gene synthesis strategies (Bayer et al., 2009). To verify the proper assembly of gene sequences, users might employ the multiple sequence alignment capabilities of Geneious. Although Geneious offers a free version with basic functionality, a license is required to use its advanced capabilities. Similar commercial offerings are **VectorNTI** (Lu, 2004) and **DNAStar Lasergene** (DNAStar, 2011). Researchers looking for a free DNA editor should consider **A Plasmid Editor** (ApE; Davis, 2011). Though ApE lacks automated assembly tools and has limited sequence alignment capabilities, it does excel in sequence visualization. Particularly, the ability to draw on information stored in a library to quickly highlight sequence features on a plasmid simplifies the process of searching DNA constructs for regions of interest.

4.2. Nucleic acid and protein synthesis

To incorporate exogenous DNA into an expression cassette, metabolic engineers may need to synthesize primers for recombinant DNA techniques or to synthesize an entire gene sequence. Researchers looking to efficiently synthesize a whole gene may

turn to **DNAWorks** (Hoover and Lubkowski, 2002), a web and Windows-based program that automates the design of short oligonucleotide sequences. When combined in a PCR reaction mixture, these oligonucleotides anneal to create a double-stranded DNA sequence. At the very least, DNAWorks requires a nucleotide sequence as well as basic information on the PCR reaction conditions in order to report a set of oligonucleotides that yields the desired gene product. The design of oligonucleotides may be further optimized using DNAWorks' ability to avoid restriction endonuclease sites or user-defined sequences. Uracil-Specific Excision Reagent fusion provides an efficient alternative to restriction–ligation or PCR-based cloning methods, and **PHUSER** (Olsen et al., 2011) is a web-based package that helps scientists quickly design primers for this molecular assembly method. Once a researcher provides the sequence for any number of DNA fragments to be combined and specifies a USER cassette, PHUSER suggests unique fusion tails that ensure proper assembly of the fragments. Both DNAWorks and PHUSER's design algorithms consider primer properties including length, GC content, melting temperature, and sequence homology.

4.3. Sequence mutation

Metabolic engineers may find that the best approach to alter enzyme concentrations is through targeted mutations to DNA or RNA sequences that encode proteins. For example, the stability of an RNA transcript can be altered by flanking the gene coding region with structures that reduce or enhance the rate of transcript degradation by RNases. Changing the degradation rate leads to different steady-state protein concentrations, and this strategy has been demonstrated as an effective way to obtain differential expression of polycistronic genes integral to a metabolic pathway (Smolke et al., 2000). **NuPack** (Zadeh et al., 2011) and **Vienna RNA Websuite** (Gruber et al., 2008) are two computational tools that excel in RNA structure design and prediction. Metabolic engineers looking to alter transcript stability may proceed by specifying a desired structure using dot-parenthesis notation (Zadeh et al., 2011). This provides either program with enough information to suggest an arbitrary nucleotide sequence that satisfies the structural constraints. However, a more likely route to RNA structure modification would involve repeated mutagenesis of a natural sequence until a structural element is added or removed. When supplied with a nucleotide sequence, both computational tools produce a structural prediction based on minimum free energy calculations and base pairing probabilities (McCaskill, 1990; Hofacker et al., 1994; Mathews et al., 1999). The tools illustrate the resulting structure and allow users to emphasize properties such as the nucleotide sequence or the probability of assuming a particular conformation, which helps to simplify the process of repeatedly refining the structure. Both NuPack and Vienna RNA Websuite are open-source and provide web-based implementations.

Modifying the ribosomal binding site (RBS) of an mRNA transcript affects the rate of translation initiation and presents researchers with an alternative approach for controlling enzyme production at the RNA level. **RBS Calculator** (Salis et al., 2009) is a web-based application that correlates translation initiation rates with RBS sequences. Metabolic engineers seeking to tune enzyme concentrations can employ this tool's *de novo* design features. A protein coding sequence, desired translation initiation rate, and host organism are required before RBS calculator can suggest a sequence. As researchers may prefer to modify an existing RBS, one valuable feature of RBS calculator is its ability to create RBSs while considering user-specified sequence constraints.

Altering the genetic code of an exogenous protein to take advantage of a host organism's codon usage bias represents a useful DNA sequence mutation strategy. The effectiveness of codon optimization has been demonstrated in many studies, including one that increased production of resveratrol in *S. cerevisiae* through mutations to the DNA sequence encoding tyrosine ammonia lyase derived from *Rhodobacter sphaeroides* (Wang et al., 2011). **Gene Designer** (Villalobos et al., 2006) is a popular, free tool for codon optimization. Gene Designer can suggest a DNA sequence once users have provided either a nucleotide or amino acid sequence and a codon usage table for a particular organism. In making its prediction, Gene Designer's codon optimization module considers four major properties: codon usage, restriction endonuclease sites, sequence homology, and mRNA secondary structure. An additional capability of this tool that metabolic engineers may enjoy is the ability to specify custom sequence constraints to minimize or maximize regions of homology with respect to a reference sequence, and this feature has been used for applications such as RNAi (Kumar et al., 2006) or gene recombination (Farrow and Arnold, 2010).

5. Metabolic flux analysis

Maximal cellular production of an interesting compound is often the defining goal of metabolic engineering. Strategies that route cellular resources away from non-essential pathways in an organism are more likely to result in the desired production goal. Mutant strains are relatively easy to experimentally construct; however, the complexity of metabolic networks makes systematic evaluation of non-intuitive knockout strategies impractical without computational assistance. Metabolic flux analysis (MFA) is the study and determination of metabolic fluxes in vivo (Stephanopoulos et al., 1998; Stephanopoulos, 1999) with the goal of maximizing product formation. Since intracellular fluxes are essentially invisible, they are inferred through stoichiometric relationships with observable processes. In the case of constraint-based approaches, intracellular fluxes are determined from measureable rates of metabolite uptake and secretion in growth medium. In the case of isotopic flux analysis, growth substrates labeled with ^{13}C are used to trace fluxes through a network.

5.1. Constraint-based analysis

Taken together, constraint-based analyses such as **FBA** (Watson, 1984; Varma and Palsson, 1994; Orth et al., 2010), **MOMA** (Segrè et al., 2002), and **ROOM** (Shlomi et al., 2005), help the metabolic engineer strategize a reasonable course of targeted manipulations that lead to a desired phenotype. Understanding the general principles of and biological interpretation from these methods provide an extremely powerful advantage to metabolic engineers. FBA is good at predicting optimal growth and yield of an organism given that there is sufficient time to adapt or evolve to selective conditions. MOMA is good at predicting sub-optimal growth immediately after a reaction knockout. The MOMA hypothesis implicitly assumes that the regulatory regime imposing the wild-type flux distribution continues to exercise control over metabolism immediately following a knockout. In this method, flux is re-routed into other pathways, and MOMA distributes the disturbance throughout the entire system. Consequently, MOMA is an improvement over FBA for lethality prediction (Segrè et al., 2002). The ROOM algorithm helps to identify flux phenotypes that are biologically plausible. It estimates the adaptation cost associated with adaptation of the native regulatory regime in the parent strain to a more optimal configuration in the target strain. The FBA solution is best regarded as a theoretical limit approached by the ROOM solution.

The **COBRA 2.0** (Schellenberger et al., 2011) toolbox for Matlab and **OptFlux** (Rocha et al., 2010) platforms implement constraint-based procedures that suggest knockout strategies in order to optimize metabolite production objectives that are coupled with cell growth objectives. Both platforms implement the **OptKnock** (Burgard et al., 2003) algorithm, which searches exhaustively for the best strategy given a pre-defined number of knockouts. The global search executed by OptKnock has some reported drawbacks in computational efficiency (Lun et al., 2009; Rocha et al., 2010); however, these can be partly circumvented by first simplifying the metabolic model. In addition, both tools feature heuristic knockout optimization algorithms that improve computational speed and, unlike OptKnock, are not constrained to a fixed number of knockouts (Patil et al., 2005; Lun et al., 2009). While heuristic methods do not guarantee the best possible strategy, or even the same strategy on successive runs, they have been demonstrated to produce results comparable to OptKnock for practical purposes (Brochado et al., 2010). A particular strength of OptFlux's interface is its flexibility in specifying different biological and production objectives, while the COBRA toolbox is more rigid in its interface with underlying linear programming solvers. Additional tools that implement constraint-based algorithms include **CellNetAnalyzer** (Klamt et al., 2007), **FASIMU** (Hoppe et al., 2011), and the **Systems Biology Research Toolbox** (Wright and Wagner, 2008).

Flux balance analysis and related constraints based modeling techniques have been used to develop a number of interesting approaches for metabolic engineering, including targeted knockouts and pathway prospecting. Constraint-based modeling has also been used in bioprocess applications for optimization of media composition (Varma and Palsson, 1994), pH balance (Motamedian and Naeimpoor, 2011), aerobic versus anaerobic switch times (Hjersted and Henson, 2006), and microbial cocultures (Hanly and Henson, 2011).

5.2. Isotopic flux analysis

While constraint-based analyses can suggest promising knockouts, the effect of genetic modification on flux distribution can range from virtually no change to altogether unexpected side effects on control mechanisms. ^{13}C flux analysis is helpful in diagnosing these effects, especially in conjunction with global expression data (Sauer, 2006). Currently available ^{13}C flux analysis platforms can be classified according to whether they implement global or local methods of inference.

13CFLUX (Wiechert et al., 2001) uses global nonlinear optimization to fit observed ^{13}C labeling patterns to a set of balance equations derived from a network model. While commonly used, the original version of 13CFLUX has a number of issues with regard to availability, accessibility, and computational efficiency (Quek et al., 2009). The latest version, **13CFLUX2** (www.13cflux.net), is greatly improved and includes faster optimization algorithms, a shift towards a new data file format, and flexible processing of spectra for MS, MS/MS, 1 H-NMR, and 13C –NMR. While 13CFLUX2 employs a command-line interface, **Omix** (Droste et al., 2011) can be used to visualize graphical models. Additionally, the recently released **OpenFlux** platform for Matlab (Quek et al., 2009) integrates many powerful ^{13}C flux analysis tools while using a GUI-interface. Furthermore, COBRA and OptFlux possess ^{13}C flux analysis capabilities.

In contrast, **FiatFlux** (Zamboni et al., 2005) performs local flux ratio analysis to determine relative fluxes at branch points. This method of inference tends to minimize errors due to inaccuracies in the network model. However, the algebraic equations that relate labeling patterns to flux ratios are usually manually derived. As a consequence, FiatFlux is currently preconfigured to calculate flux ratios for only a few organisms grown on glucose.

6. Pathway prospecting

Researchers often prefer to use model organisms with well-studied networks for metabolic engineering applications. This is often due to the incredible amount of effort required to construct genome-scale metabolic network models or because of the lack of cell culturing and molecular cloning techniques for a particular organism. An important future direction for metabolic engineering involves developing techniques to effectively search for and identify exogenous reaction pathways that can be introduced into standard production systems such as E. coli. Specifically, how can researchers systematically select the appropriate set of pathways and genes from the myriad of data cataloged in various biological databases to endow existing organisms with new functionality? In its simplest form, this idea is no different than achieving heterologous enzyme expression and effective computational and experimental strategies already exist. However, complex situations which require larger metabolic pathway grafts present a more difficult task.

Consider a study that investigated the capacity for alkane production in E. coli (Schirmer et al., 2010). Alkanes are a major component in jet diesel and gasoline, and this work proceeded by comparing alkane biosynthesis pathways in cyanobacteria. The **Integrated Microbial Genomes** (**IMG**; Markowitz et al., 2006) system was used to compare genomes from various strains of cyanobacteria. First, a set of necessary genes was developed by identifying common genes among cyanobacteria that produced alkanes. Next, genes that were shared between strains that did not produce alkanes were subtracted from the list of necessary genes. The remaining genes were identified as essential candidates for alkane biosynthesis, and they were shown to be sufficient for alkane production in E. coli.

Methods that automate pathway prospecting increase the practicality of this technique, and automation strategies will likely involve comparing biochemical reactions obtained from various databases. **OptStrain** (Pharkya and Maranas, 2002) is an algorithm that searches various databases to discover, analyze, and select pathways that lead to production of a target molecule. Under the OptStrain framework, maximum theoretical yields are sought; however, it simultaneously considers minimizing the number of exogenous reactions. Additionally, it integrates with the **OptKnock** (Burgard et al., 2003) programming framework to further increase target production through target gene deletions.

7. Post-structural network analysis

Although at the present time the bulk of computational analysis on metabolic networks focuses on properties of the structure of networks (see Section 4), there is a small but growing body of literature that is using dynamical approaches. Dynamics analysis incorporates network kinetics, in addition to network structure, to understand and assist in engineering metabolism (Steuer and Junker, 2008; Chandran et al., 2009b; Sauro, 2009). The types of work conducted in this area can be divided into two broad areas: using dynamical models to understand the design of metabolic networks; and using dynamical models to find suitable targets for metabolic engineering.

7.1. Understanding the design of metabolism

There is still much to be learned regarding the operational characteristics of metabolic networks (Fell, 1996). For example, it is common in all metabolic pathways to find feed-forward and feedback regulatory signals. In certain scenarios, this may include nested regulation that involves coordinated positive and negative interactions between components in both protein and gene

regulatory networks. In the majority of cases we only understand the role of these interactions at a superficial level.

One recent study combined experimentation with dynamic modeling to yield a compelling explanation for the existence of nested feedback loops in the highly branched *Arabidopsis* amino acid biosynthesis pathway (Curien et al., 2009). This work discovered that allosteric interactions in the biosynthesis pathway served to maintain a high independence between competing fluxes. This ensures that increased demand in one branch does not lead to a loss of flux in another branch. In addition, the work showed that the various enzyme isoforms are not redundant, and that threonine was the most sensitive variable in the pathway which implies additional roles for threonine in regulating elsewhere. In this example **COPASI** (Mendes et al., 2009) was used as the primary simulation and analysis tools. COPASI is open-source, features a strong capacity for optimization and parameter fitting, and is commonly used to simulate deterministic as well as stochastic models. COPASI is an excellent tool for this kind of study because it includes such a wide variety of dynamical analyses, including metabolic control analysis capabilities (Kacser and Burns, 1973). COPASI was used to carry out a number of tasks in the aforementioned study, which included carrying out time course simulations, adjusting parameter values for simulation, confirming that the steady state of the system was stable, metabolic control analysis, and analyzing how demand at various points in the network influenced pathway flux. It is worth mentioning that COPASI's ability to export SBML enabled other researchers to easily gain access to the models distributed from the highlighted study.

7.2. Locating enzyme targets

The other useful role that dynamic modeling can play is determining the most likely enzyme targets for altering expression levels. Discovering flux sensitivity of enzymatic steps in a pathway presents engineers with a powerful method for understanding control over a metabolic network.

Two examples illustrate this application: the first involves a non-intuitive discovery from a study of the serine biosynthesis pathway; the second uses dynamical modeling to locate suitable drug targets in the metabolic pathways of a parasite. The first study used dynamical systems analysis on a serine biosynthesis pathway in mammalian cells to show that, contrary to expectations, the last step in the pathway exerted the most control over flux (Fell and Snell, 1988). Computational analysis using any of the methods discussed in previous sections of this review would not have uncovered this result with the resultant misdirection of engineering resources. Traditional approaches would have assumed that the committed step had most of the control, and re-engineering the first step in the pathway would have had little effect on improving serine output. Although flux control is influenced to some degree by network structure (Kacser and Burns, 1973), kinetic considerations play a much more significant role. A second example used dynamical analysis to identify targets that would disrupt *Trypanosoma brucei* physiology (Bakker et al., 2000). The work was centered on modeling the parasite's ATP generation pathway, and it used sensitivity analysis to discover possible drug targets. Ultimately, the authors found that most of the pathway metabolites are insensitive because of a conserved phosphate pool and that there was only limited scope to affect flux.

Studies like these can be aided by tools that support sensitivity analysis, metabolic control analysis, and dynamics simulation. The **Systems Biology Workbench** (Sauro et al., 2004; Bergmann and Sauro, 2006), **COPASI**, or **PySCeS** (Olivier et al., 2005) are three common tools that can be used for these types of dynamical analyses. To perform sensitivity analysis within any of these tools,

metabolic engineers would begin by simulating a network model to obtain steady-state values. The computational tools allow users to either manually or programmatically alter kinetic parameters, and upon completion, users may observe the change in steady-state substrate concentrations with respect to the magnitude of change for various parameters. Additionally, the analysis capabilities of these tools have been integrated into various other tools, such as TinkerCell (Chandran et al., 2009), that attempt to predict optimal targets for pathway control.

The major obstacle in all dynamic modeling is the lack of kinetic data (Schallau and Junker, 2010). This is the reason why most dynamical models are small and relatively uncommon. However, efforts have started to consider building models using generalized rate laws (Liebermeister et al., 2010; Sauro, 2011) when detailed kinetic information is lacking. Such approaches could conceivably be applied to genome scale dynamical models. In addition a linear analysis can be carried out if thermodynamic data and metabolite concentrations are available (Sauro, 2011). Such an analysis would provide a first approximation to the expected dynamics and in particular allow the computation of the flux and metabolite sensitivities to enzyme expression changes (Steuer and Junker, 2008).

8. Culture optimization

Introducing foreign pathways into a host for heterologous expression often results in poor or no growth for a variety of reasons. These reasons include non-suitable hosts, low substrate availability, a shortage or accumulation of intermediate metabolites, competing metabolic pathways, or restricted storage capacity for end products (Klein-Marcuschamer et al., 2007). Since altering metabolism may slow growth, metabolic engineers may need to consider the effect of culture conditions on maximum product yields. Considering every variable related to culture conditions is daunting; however, certain tools can help optimize growth and direct flux towards a product of interest. Most of these tools assess growth under various culture conditions by applying flux analysis methods to genome-scale metabolic models with the primary intention of maximizing growth rate. As discussed earlier, FBA platforms can be used towards this end; however, in this section we will highlight tools that were employed with the sole intention of studying the effect of various substrates on cellular growth.

GrowMatch (Kumar and Maranas, 2009) was used to improve growth predictions in the *in silico E. coli* model iAF1260. GrowMatch increased the consistency of the *in silico* model by considering experimental gene essentiality data from a comprehensive collection of single-mutant *E. coli* K-12 BW25113 cells exposed to different substrate conditions. Through *in silico* and in vivo data comparisons, GrowMatch automates the inclusion of missing growth predictions by identifying the minimal set of modifications that enable biomass formation. Modifications may include modifying biomass equations, finding possible alternate genes that carry-out the function of the deleted gene, or adding new functionalities to the model. Likewise, it removes erroneous growth predictions by identifying ways to negate biomass formation, such as suggesting different suppression interactions.

BioMet Toolbox is a web-accessible tool suite that can be used to optimize growth rates, substrate uptake rates, or metabolic production rates (Cvijovic et al., 2010). BioMet Toolbox's BioOpt and Reporter Features modules integrate flux balance analysis methods with collections of transcription and protein interaction data to predict maximum yields under specific culture conditions. The authors performed a case study to estimate the flux for growth during aerobic fermentation in yeast. Simulations were

performed using either glucose or ethanol as a carbon source, and predictions for metabolite production and consumption rates were in good agreement with experimental findings.

8.1. Future challenges and developments

Despite the many software tools that exist to support metabolic engineering endeavors, there are still significant metabolic engineering challenges that computational tools cannot adequately address. A prerequisite to developing useful software tools that address these types of challenges is improving upon the theoretical understanding within those areas of metabolic engineering research. As theory and expertise increase, software tools that support metabolic pathway design will follow. These tools will likely incorporate and interpret new types of data when suggesting design predictions.

For example, one remaining challenge involves considering toxic intermediates during pathway optimization. Software tools such as **metaP-server** (Kastenmüller et al., 2011) and **MeltDB** (Neuweger et al., 2008) perform statistical analysis on metabolomics data in an effort to link phenotypes with metabolite levels. Information exchange between software tools that interpret metabolomics data and tools that perform metabolic flux analysis represents a potential solution to address this challenge. Another compelling task entails designing robust pathways that operate reliably under a variety of extracellular conditions. Gene network engineering approaches that implement protein expression control strategies may prove useful in ensuring metabolic networks

Table 1
Software tools used in metabolic engineering applications.

Name	Tasks	License	Accessibility
13CFLUX2	MFA	Free non-commercial	UNIX/Linux
A Plasmid Editor (ApE)	DNA visualization, Nucleic acid design	Free	Cross-Platform
Arcadia	Reaction network visualization	GPL	Cross-Platform
BiGG	Metabolic network reconstruction	Free non-commercial	Online
BioMet Toolbox	Constraints-based modeling	Free	Online, Windows
BioModelsDB (Le Novère et al., 2006)	Metabolic network reconstruction	Free	Online
BioPax	Annotation	Free	N/A
BioTapestry	Genetic network construction and analysis	Free	Cross-Platform
BLAST	Comparative sequence analysis	Free	Online, Cross-Platform
Cell Illustrator	Reaction network visualization and design	Free, Closed source	Online
CellDesigner	Reaction network visualization and design	Free, Closed source	Cross-Platform
CellNetAnalyzer	Constraint-based modeling, MFA, Network analysis	Free academic, Requires Matlab	Cross-Platform
COBRA 2.0	Constraint-based Modeling, MFA, Network analysis	GNU GPLv3	Cross-Platform
COPASI	Mathematical analysis	Artistic License 2.0	Cross-Platform
Cytoscape	Interaction network visualization	GNU LGPL	Cross-Platform
DNA 2.0 Gene Designer	Codon optimization	Free, Closed source	Cross-Platform
DNAStar Lasergene	DNA visualization, Nucleic acid design	Academic, Commercial	Cross-Platform
FASIMU	Constraint-based modeling, MFA	GNU GPL	Cross-Platform
FiatFlux	MFA	Free academic, Requires Matlab	Cross-Platform
Geneious	DNA visualization, Nucleic acid design	Free limited, Academic, Commercial	Cross-Platform
GenoCAD	DNA visualization, Nucleic acid design	Apache 2.0	Online
GLAMM	Reaction network visualization	BSD 2	Online
GO	Annotation	Free	N/A
GraphViz	Interaction network visualization	Eclipse Public License	Cross-Platform
GrowMatch	Optimize culture conditions	Source code available to academic users	Cross-Platform
HelixWeb DNA Works	Gene synthesis	Free, Closed source	Online, Windows
IMG	Comparative sequence analysis, Annotation	Free, Closed source	Online
JDesigner	Reaction network visualization and design	BSD 2	Windows
KAAS	Metabolic network reconstruction	Free	Online
KEGG Pathway	Metabolic network reconstruction	Free web, Licensed download	Online
MetaCyc	Metabolic network reconstruction	Free agreement	Online
MetRxn	Metabolic network reconstruction	Free	Online
ModelSEED	Metabolic network reconstruction	Free	Online
NuPack	Nucleic acid structure analysis	Free, Open source	Online
Omix	Reaction network visualization	Free non-commercial, Closed source	Cross-Platform
OpenFLUX	MFA	GNU GPL, Requires Matlab	Cross-Platform
OptFlux	Constraint-based modeling, MFA, Network analysis	GNU GPLv3	Cross-Platform
OptKnock	Constraints-based modeling	Free, Requires Matlab	Cobra toolbox 2.0
OptStrain	Pathway prospecting	Free	Available by request
PathwayTools	Metabolic network model analysis	Free non-commercial	Cross-Platform
PHUSER	Primer design	Free	Online
PySCeS	Dynamic simulation	BSD 2	Cross-Platform
RBS Calculator	Nucleic acid design, Expression optimization	Free non-commercial	Online
Reactome (Croft et al., (2010))	Metabolic network reconstruction	Free	Online
SBGN	Network visualization	Free	N/A
SBML	Network reconstruction and visualization	Free	N/A
SBO	Annotation	Free	N/A
SBW	Dynamic simulation	BSD 2	Cross-Platform
SL Finder	Optimize culture conditions	Source code available to academic users	Cross-Platform
Systems Biology Research Tool	Constraint-based modeling, MFA, Network analysis	GNU GPLv2	Cross-Platform
Systrip	Interaction network visualization	GNU LGPL	Cross-Platform
TinkerCell	Model visualization and analysis	BSD 2	Cross-Platform
Vanted	Reaction network visualization	GNU GPLv2	Cross-Platform
VectorNTI	DNA visualization, Nucleic acid design	Academic, Commercial	Cross-Platform
Vienna RNA Websuite	Nucleic acid structure analysis	Free, Open source	Online
yEd	Interaction network visualization	Free, Closed source	Cross-Platform

perform consistently across a range of experimental conditions. One major focus of work in the field of synthetic biology involves engineering complex cellular behavior via assembly and expression of genes that encode well-characterized biological components. While the link between current synthetic biology tools and metabolic pathway engineering is tenuous, it is very possible that future tools will be able to suggest gene network designs that intelligently regulate the expression of constituents in metabolic pathways.

9. Conclusion

The diversity of metabolic engineering tasks has motivated an expansive collection of computational tools that support experimental techniques. In this review, we have presented a variety of tools that assist scientists in successfully modifying cellular metabolism. Table 1 provides a summary of the tools discussed in this review, includes the types of tasks they perform, and notes licensing and platform requirements. Collectively, these computational tools provide a powerful framework to enhance design and analysis aspects of metabolic engineering.

Acknowledgments

Appendix A. Supplementary material

Supplementary data associated with this article can be found in the online version at doi:10.1016/j.ymben.2012.03.001.

References

Altschul, S.F., Gish, W., Miller, W., Myers, E.W., Lipman, D.J., 1990. Basic local alignment search tool. J. Mol. Biol. 215, 403–410.
Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., et al., 2000. Gene ontology: tool for the unification of biology. The gene ontology consortium. Nat. Genet. 25, 25–29.
Auber, D., 2003. Tulip: A huge graph visualization framework. In: Mutzel, P., Jünger, M. (Eds.), Math Visual. Springer-Verlag, pp. 105–126.
Bakker, B.M., Mensonides, F.I., Teusink, B., Hoek, P., van, Michels, P.A., Westerhoff, H.V., 2000. Compartmentation protects trypanosomes from the dangerous design of glycolysis. Proc. Natl. Acad. Sci. USA 97, 2087–2092.
Bates, J.T., Chivian, D., Arkin, Adam P., 2011. GLAMM: Genome-Linked Application for Metabolic Maps. Nucleic Acids Res. 39, W400–W405.
Bayer, T.S., Widmaier, D.M., Temme, K., Mirsky, E.A., Santi, D.V., Voigt, C.A., 2009. Synthesis of methyl halides from biomass using engineered microbes. J. Am. Chem. Soc. 131, 6508–6515.
Bergmann, F.T., Sauro, H.M., 2006. SBW-a modular framework for systems biology. In: Proceedings of the 38th conference on Winter simulation, pp. 1637–1645.
Brochado, A.R., Matos, C., Møller, B.L., Hansen, J., Mortensen, Uffe H, Patil, Kiran Raosaheb, 2010. Improved vanillin production in baker's yeast through in silico design. Microb. Cell Fact. 9, 84.
Burgard, A.P., Pharkya, P., Maranas, C.D., 2003. Optknock: a bilevel programming framework for identifying gene knockout strategies for microbial strain optimization. Biotechnol. Bioeng. 84, 647–657.
Cai, Y., Wilson, M.L., Peccoud, J., 2010. GenoCAD for iGEM: a grammatical approach to the design of standard-compliant constructs. Nucleic Acids Res. 38, 2637–2644.
Caspi, R., Foerster, H., Fulcher, C.A., Kaipa, P., Krummenacker, M., Latendresse, M., et al., 2008. The MetaCyc Database of metabolic pathways and enzymes and

the BioCyc collection of Pathway/Genome Databases. Nucleic Acids Res. 36, D623–D631.
Caspi, R., Altman, T., Dale, J.M., Dreher, K., Fulcher, C.A., Gilham, F., et al., 2010. The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. Nucleic Acids Res. 38, D473–D479.
Chance, B., 1943. The Kinetics of the enzyme-substrate compound of peroxidase. J. Biol. Chem. 151, 553–577.
Chance, B., Greenstein, D.S., Higgins, J., Yang, C.C., 1952. The mechanism of catalase action. II. Electric analog computer studies. Arch. Biochem. 37, 322–339.
Chance, B., Garfinkel, D., Higgins, J., Hess, B., 1960. Metabolic control mechanisms. 5. A solution for the equations representing interaction between glycolysis and respiration in ascites tumor cells. J. Biol. Chem. 235, 2426–2439.
Chandran, D., Bergmann, F.T., Sauro, H.M., 2009a. TinkerCell: modular CAD tool for synthetic biology. J. Biol. Eng. 3, 19.
Chandran, D., Copeland, W.B., Sleight, S.C., Sauro, H.M., 2009b. Mathematical modeling and synthetic biology. Drug Discov. Today: Dis. Models 5 (4), 299–309.
Chandran, D., Bergmann, F.T., Sauro, H.M., 2010. Computer-aided design of biological circuits using TinkerCell. Bioeng. Bugs 1, 274–281.
Cohen, S.N., 1973. Construction of biologically functional bacterial plasmids in vitro. Proc. Natl. Acad. Sci. USA 70, 3240–3244.
Croft, D., et al., 2010. Reactome: a database of reactions, pathways and biological processes. Nucleic Acids Res. 39, D691–D697.
Curien, G., Bastien, O., Robert-Genthon, M., Cornish-Bowden, A., Cárdenas, M.L., Dumas, R., 2009. Understanding the regulation of aspartate metabolism using a model based on measured kinetic parameters. Mol. Syst. Biol. 5, 271.
Cvijovic, M., Olivares-Hernández, R., Agren, R., Dahr, N., Vongsangnak, W., Nookaew, I., Patil, Kiran Raosaheb, Nielsen, J., 2010. BioMet Toolbox: genome-wide analysis of metabolism. Nucleic Acids Res. 38, W144–W149.
Czar, M.J., Cai, Y., Peccoud, J., 2009. Writing DNA with GenoCAD. Nucleic Acids Res. 37, W40–W47.
DNAStar, 2011. DNAStar Lasergene. ⟨http://www.dnastar.com/⟩ (Accessed May 26, 2011).
Davis, M.W., 2011. A plasmid Editor. ⟨http://biologylabs.utah.edu/jorgensen/wayned/ape/⟩.
DeJongh, M., Formsma, K., Boillot, P., Gould, J., Rycenga, M., Best, A., 2007. Toward the automated generation of genome-scale metabolic networks in the SEED. BMC Bioinf. 8, 139.
DeJongh, M., Bockstege, B., Frybarger, P., Hazekamp, N., Kammeraad, J., McGeehan, T., 2011. CytoSEED: a Cytoscape plugin for viewing, manipulating and analyzing metabolic models created by the Model SEED. Bioinformatics, 10.1093/bioinformatics/btr719.
Dehal, P.S., Joachimiak, M.P., Price, M.N., Bates, J.T., Baumohl, J.K., Chivian, D., et al., 2010. MicrobesOnline: an integrated portal for comparative and functional genomics. Nucleic Acids Res. 38, D396–D400.
Demir, E., Cary, M.P., Paley, S., Fukuda, K., Lemer, C, Vastrik, I., et al., 2010. The BioPAX community standard for pathway data sharing. Nat. Biotechnol. 28, 935–942.
Dien, S.J., Van, Strovas, T., Lidstrom, M.E., 2003. Quantification of central metabolic fluxes in the facultative methylotroph methylobacterium extorquens AM1 using 13C-label tracing and mass spectrometry. Biotechnol. Bioeng. 84, 45–55.
Droste, P., Miebach, S., Niedenführ, S., Wiechert, W., Nöh, K., 2011 Visualizing multi-omics data in metabolic networks with the software Omix—A case study. Biosystems, 105 (2), pp. 154–161.
Drummond, A.J., Ashton, B., Buxton, S., Cheung, M., Cooper, A., Duran, C., Field, M., Heled, J., Kearse, M., Markowitz, S., Moir, R., Stones-Havas, S., Sturrock, S., Thierer, T., Wilson, A., 2011. Geneious v5.4. Available from ⟨http://www.geneious.com/⟩.
Durot, M., Bourguignon, P.-Y., Schachter, V., 2009. Genome-scale models of bacterial metabolism: reconstruction and applications. FEMS Microbiol. Rev. 33, 164–190.
Eilbeck, K., Lewis, S.E., Mungall, C.J., Yandell, M., Stein, L., Durbin, R., Ashburner, Michael, 2005. The sequence ontology: a tool for the unification of genome annotations. Genome Biol. 6, R44.
Farrow, M.F., Arnold, F.H., 2010. Combinatorial recombination of gene fragments to construct a library of chimeras. Current protocols in protein science/editorial board, John E. Coligan, et al. Chapter 26, Unit 26.2.
Fell, D., 1996. Understanding the Control of Metabolism. Portland Press Ltd, London.
Fell, D.A., Snell, K., 1988. Control analysis of mammalian serine biosynthesis. Feedback inhibition on the final step. Biochem. J. 256, 97–101.
Funahashi, A., Morohashi, M., Kitano, H., Tanimura, N., 2003. CellDesigner: a process diagram editor for gene-regulatory and biochemical networks. Biosilico 1, 159–162.
Funahashi, A., Matsuoka, Y., Jouraku, A., Morohashi, M., Kikuchi, N., Kitano, Hiroaki, 2008. CellDesigner 3.5: a versatile modeling tool for biochemical networks. Proc. IEEE 96, 1254–1265.
Galdzicki, M., Rodriguez, C., Chandran, D., Sauro, H.M., Gennari, J.H., 2011. Standard Biological Parts Knowledgebase Schönbach C. (Ed.). PLoS ONE 6, e17005.
Gansner, E.R., North, S.C., 2000. An open graph visualization system and its applications to software engineering. Software Pract. Exp. 30, 1203–1233.
Gruber, A.R., Lorenz, R., Bernhart, S.H., Neuböck, R., Hofacker, Ivo L., 2008. The Vienna RNA websuite. Nucleic Acids Res. 36, W70–W74.
Hanly, T.J., Henson, M.A., 2011. Dynamic flux balance modeling of microbial co-cultures for efficient batch fermentation of glucose and xylose mixtures. Biotechnol. Bioeng. 108, 376–385.

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, 977–982.

Hjersted, J.L., Henson, M.A., 2006. Optimization of fed-batch Saccharomyces cerevisiae fermentation using dynamic flux balance models. Biotechnol. Progr. 22, 1239–1248.

Hofacker, I.L., Fontana, W., Stadler, P.F., Bonhoeffer, L.S., Tacker, M., Schuster, P., 1994. Fast folding and comparison of RNA secondary structures. Monatsh. Chem. 125, 167–188.

Hoover, D.M., Lubkowski, J., 2002. DNAWorks: an automated method for designing oligonucleotides for PCR-based gene synthesis. Nucleic Acids Res. 30, e43.

Hoppe, A., Hoffmann, S., Gerasch, A., Gille, C., Holzhutter, H.-G., 2011. FASIMU: flexible software for flux-balance computation series in large metabolic networks. BMC Bioinf. 12, 28.

Hucka, M., Finney, A., Sauro, H.M., Bolouri, H., Doyle, J.C., Kitano, H., et al., 2003. The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. Bioinf. (Oxford, England) 19, 524–531.

Johnson, M., Zaretskaya, I., Raytselis, Y., Merezhuk, Y., McGinnis, S., Madden, T.L., 2008. NCBI BLAST: a better web interface. Nucleic Acids Res. 36, W5–W9.

Kacser, H., Burns, J.A., 1973. The control of flux. Symp. Soc. Exp. Biol. 27, 65–104.

Kanehisa, M., Goto, S., 2000. KEGG: Kyoto Encyclopedia of Genes and Genomes. Nucleic Acids Res. 28, 27–30.

Kanehisa, M., Goto, S., Hattori, M., Aoki-Kinoshita, K.F., Itoh, M., Kawashima, S., Katayama, T., Araki, M., Hirakawa, M., 2006. From genomics to chemical genomics: new developments in KEGG. Nucleic Acids Res. 34, D354–D357.

Karimi, M., Depicker, A., Hilson, P., 2007. Recombinational cloning with plant gateway vectors. Plant Physiol. 145, 1144–1154.

Karp, P.D., Paley, S., Romero, P., 2002. The pathway tools software. Bioinformatics 18, S225–S232.

Karp, P.D., Caspi, R., 2011. A survey of metabolic databases emphasizing the MetaCyc family. Arch. Toxicol.. doi:10.1007/s00204-011-0705-2.

Karp, P.D., Paley, S.M., Krummenacker, M., Latendresse, M., Dale, J.M., Lee, T.J., et al., 2010. Pathway Tools version 13.0: integrated software for pathway/genome informatics and systems biology. Brief Bioinf. 11, 40–79.

Kastenmüller, G., Römisch-Margl, W., Wägele, B., Altmaier, E., Suhre, K., 2011. metaP-Server: A web-based metabolomics data analysis tool. J. Biomed. Biotechnol.. doi:10.1155/2011/839862.

Klamt, S., Saez-Rodriguez, J., Gilles, E.D., 2007. Structural and functional analysis of cellular networks with CellNetAnalyzer. BMC. Syst. Biol. 1, 2.

Klein-Marcuschamer, D., Ajikumar, P.K., Stephanopoulos, G., 2007. Engineering Microb Cell Fact for biosynthesis of isoprenoid molecules: beyond lycopene. Trends Biotechnol. 25, 417–424.

Klukas, C., Schreiber, F., 2010. Integration of omics data and networks for biomedical research with VANTED. J. Int. Bioinf. 7, 112.

Kohl, M., Wiese, S., Warscheid, B., 2011. Cytoscape: software for visualization and analysis of biological networks. Methods. Mol. Biol. 696, 291–303.

Kortemme, T., Baker, D., 2004. Computational design of protein-protein interactions. Curr. Opin. Chem. Biol. 8 (1), 91–97.

Kumar, A., Suthers, P., Maranas, C.D., 2011. MetRxn□: reaction and metabolite standardization and congruency across databases and genome-scale metabolic models. In: GTL Conference. pp. 1–2. 〈https://www.orau.gov/gtl2011/abstracts/maranas_costas.pdf〉.

Kumar, D., Gustafsson, C., Klessig, D.F., 2006. Validation of RNAi silencing specificity using synthetic genes: salicylic acid-binding protein 2 is required for innate immunity in plants. Plant J. 45, 863–868.

Kumar, V.S., Maranas, C.D., 2009. GrowMatch: an automated method for reconciling in silico/in vivo growth predictions. Plos. Comput. Biol. 5, e1000308.

Le Novère, N., Bornstein, B., Broicher, A., Courtot, M., Donizelli, M., Dharuri, H., et al., 2006. BioModels Database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. Nucleic Acids Res. 34, D689–D691.

Le Novère, N., Hucka, M., Mi, H., Moodie, S., Schreiber, F., Sorokin, A., et al., 2009. The systems biology graphical notation. Nat. Biotechnol. 27, 735–741.

Le Novère, N., 2011. System Biology Ontology. 〈www.ebi.ac.uk/sbo/〉 (Accessed June 2, 2011).

Liebermeister, W., Uhlendorf, J., Klipp, E., 2010. Modular rate laws for enzymatic reactions: thermodynamics, elasticities and implementation. Bioinformatics 26 (12), 1528–1534.

Longabaugh, W.J.R., Davidson, E.H., Bolouri, Hamid, 2009. Visualization, documentation, analysis, and communication of large-scale gene regulatory networks. Biochim. Biophys. Acta 1789, 363–374.

Lu, G., 2004. Vector NTI, a balanced all-in-one sequence analysis suite. Brief Bioinf. 5, 378–388.

Lun, D.S., Rockwell, G., Guido, N.J., Baym, M., Kelner, J.A., Berger, B., Galagan, J.E., Church, G.M., 2009. Large-scale identification of genetic design strategies using local search. Mol. Syst. Biol. 5, 296.

Markowitz, V.M., Korzeniewski, F., Palaniappan, K., Szeto, E., Werner, G., Padki, A., et al., 2006. The integrated microbial genomes (IMG) system. Nucleic Acids Res. 34, D344–D348.

Mathews, D.H., Sabina, J., Zuker, M., Turner, D.H., 1999. Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. J. Mol. Biol. 288, 911–940.

McCaskill, J.S., 1990. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. Biopolymers 29, 1105–1119.

Mendes, Pedro, Hoops, S., Sahle, S., Gauges, R., Dada, J., Kummer, Ursula, 2009. Computational modeling of biochemical networks using COPASI. Methods Mol. Biol. (Clifton, N.J.) 500, 17–59.

Moriya, Y., Itoh, M., Okuda, S., Yoshizawa, A.C., Kanehisa, M., 2007. KAAS: an automatic genome annotation and pathway reconstruction server. Nucleic Acids Res. 35, W182–W185.

Motamedian, E., Naeimpoor, F., 2011. Prediction of proton exchange and bacterial growth on various substrates using constraint-based modeling approach. Biotechnol. Bioproc. E 16, 875–884.

Nagasaki, M., Saito, Ayumu, Jeong, Euna, Li, Chen, Kojima, Kaname, Ikeda, Emi, Satoru, Miyano, 2010. Cell Illustrator 4.0: a computational platform for systems biology. In Silico Biol., 10.

Notebaart, R., Enckevort, F., van, Francke, C., Siezen, R., Teusink, Bas, 2006. Accelerating the reconstruction of genome-scale metabolic networks. BMC Bioinf. 7, 296.

Neuweger, H., Albaum, S.P., Dondrup, M., Persicke, M., Watt, T., Niehaus, K., Stoye, J., Goesmann, A., 2008. MeltDB: a software platform for the analysis and integration of metabolomics experiment data. Bioinformatics 24 (23), 2726–2732.

Ogata, H., Goto, S., Sato, K., Fujibuchi, W., Bono, H., Kanehisa, M., 1999. KEGG: Kyoto encyclopedia of genes and genomes. Nucleic Acids Res. 27, 29–34.

Olivier, B.G., Rohwer, J.M., Hofmeyr, J.-H.S., 2005. Modelling cellular systems with PySCeS. Bioinformatics (Oxford, England) 21, 560–561.

Olsen, L.R., Hansen, N.B., Bonde, M.T., Genee, H.J., Holm, D.K., Carlsen, S., et al., 2011. PHUSER (Primer Help for USER): a novel tool for USER fusion primer design. Nucleic Acids Res. 39 (Suppl 2), W61–W67.

Orth, J.D., Thiele, I., Palsson, B.Ø., 2010. What is flux balance analysis? Nat. Biotechnol. 28, 245–248.

Patil, Kiran Raosaheb, Rocha, I., Förster, J., Nielsen, J., 2005. Evolutionary programming as a platform for in silico metabolic engineering. BMC Bioinf. 6, 308.

Pharkya, P., Maranas, C.D., 2002. OptStrain: A hierarchical metabolic pathway discovery and design framework for microbial production. Production.

Quek, L.-E., Wittmann, C., Nielsen, L.K., Krömer, J.O., 2009. OpenFLUX: efficient modelling software for 13C-based metabolic flux analysis. Microb. Cell Fact. 8, 25.

Rocha, I., Maia, P., Evangelista, P., Vilaça, P., Soares, S., Pinto, J.P., Nielsen, J., Patil, Kiran R, Ferreira, E.C., Rocha, M., 2010. OptFlux: an open-source software platform for in silico metabolic engineering. BMC Syst. Biol. 4, 45.

Salis, H.M., Mirsky, E.A., Voigt, C.A., 2009. Automated design of synthetic ribosome binding sites to control protein expression. Nat. Biotechnol. 27, 946–950.

Sauer, U., 2006. Metabolic networks in motion: 13C-based flux analysis. Mol. Syst. Biol. 2, 62.

Sauro, H.,M., Hucka, M., Finney, A., Wellock, C., Bolouri, H., Doyle, J., Kitano, H., 2004. Next generation simulation tools: the systems biology workbench and BioSPICE integration. OMICS 7 (4), 355–372.

Sauro, H.M., 2008. Standards and ontologies in computational systems biology. Essays Biochem. 45, 211–222.

Sauro, H.M., 2009. Network dynamics, computational systems biology. Methods Mol. Biol. 541 (3), 269–309. ed.: Ireton, R, Montmery K, Bumgarner, R, Samudrala R and McDermott J.

Sauro, H.M., 2011. Enzyme Kinetics for Systems Biology. Ambrosius Publishing, Seattle, ISBN: 0982477317.

Schallau, K., Junker, B.H., 2010. Simulating plant metabolic pathways with enzyme-kinetic models. Plant Physiol. 152, 1763–1771.

Schellenberger, J., Park, J.O., Conrad, T.M., Palsson, B.Ø., 2010. BiGG: a biochemical genetic and genomic knowledgebase of large scale metabolic reconstructions. BMC Bioinf. 11, 213.

Schellenberger, J., Que, R., Fleming, R.M.T., Thiele, I., Orth, J.D., Feist, A.M., et al., 2011. Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox v2.0. Nat. Protoc. 6, 1290–1307.

Schirmer, A., Rude, M.A., Li, X., Popova, E., Cardayre, S.B. del, 2010. Microbial biosynthesis of alkanes. Science (New York, N.Y.) 329, 559–562.

Segrè, D., Vitkup, D., Church, G.M., 2002. Analysis of optimality in natural and perturbed metabolic networks. Proc. Natl. Acad. Sci. USA 99, 15112–15117.

Shannon, P., Markiel, A., Ozier, O., Baliga, N.S., Wang, J.T., Ramage, D., Amin, N., Schwikowski, B., Ideker, T., 2003. Cytoscape: a software environment for integrated models of biomolecular interaction networks. Genome Res. 13, 2498–2504.

Shetty, R.P., Endy, D., Knight, T.F., 2008. Engineering BioBrick vectors from BioBrick parts. J. Biol. Eng. 2, 5.

Shlomi, T., Berkman, O., Ruppin, E., 2005. Regulatory on/off minimization of metabolic flux changes after genetic perturbations. Proc. Natl. Acad. Sci. USA 102, 7695–7700.

Sleight, S.C., Bartley, B.A., Lieviant, J.A., Sauro, H.M., 2010. In-Fusion BioBrick assembly and re-engineering. Nucleic Acids Res.

Smolke, C.D., Carrier, T.A., Keasling, J.D., 2000. Coordinated, differential expression of two genes through directed mRNA cleavage and stabilization by secondary structures. Appl. Environ. Microb. 66, 5399–5405.

Stephanopoulos, G., 1999. Metabolic fluxes and metabolic engineering. Metab. Eng. 1, 1–11.

Stephanopoulos, Gregory, Nielsen, J., Aristidou, A.A., 1998. Metabolic Engineering. Principles and Methodologies. Academic Press.

Steuer, R., Junker, B.H., 2008. Computational Models of Metabolism: Stability and Regulation in Metabolic Networks. Advances in Chemical Physics, 142. John Wiley & Sons, Inc., Hoboken, NJ, USA ed S.A. Rice.

Thiele, I., Palsson, B.Ø., 2010. A protocol for generating a high-quality genome-scale metabolic reconstruction. Nat. Protoc. 5, 93–121.

Varma, A., Palsson, B.O., 1994. Stoichiometric flux balance models quantitatively predict growth and metabolic by-product secretion in wild-type Escherichia coli W3110. Appl. Environ. Microb. 60, 3724–3731.

Villalobos, A., Ness, J.E., Gustafsson, C., Minshull, J., Govindarajan, S., 2006. Gene designer: a synthetic biology tool for constructing artificial DNA segments. BMC Bioinf. 7, 285.

Villéger, A.C., Pettifer, S.R., Kell, D.B., 2010. Arcadia: a visualization tool for metabolic pathways. Bioinformatics (Oxford, England) 26, 1470–1471.

Wang, Y., Halls, C., Zhang, J., Matsuno, M., Zhang, Y., Yu, O., 2011. Stepwise increase of resveratrol biosynthesis in yeast Saccharomyces cerevisiae by metabolic engineering. Metab. Eng. 13 (5), 455–463.

Watson, M.R., 1984. Metabolic maps for the Apple II. Diabetes 12, 1–2.

Wiechert, W., Möllney, M., Petersen, S., Graaf, A.A. de, 2001. A universal framework for 13C metabolic flux analysis. Metab. Eng. 3, 265–283.

Wright, J., Wagner, A., 2008. The systems biology research tool: evolvable open-source software. BMC Syst. Biol. 2, 55.

Wrzodek, C., Dräger, A., Zell, A., 2011. KEGG translator: visualizing and converting the KEGG PATHWAY database to various formats. Bioinformatics, 1–2.

Yoon, S.H., Kim, J.E., Lee, S.H., Park, H.M., Choi, M.S., Kim, J.Y., et al., 2007. Engineering the lycopene synthetic pathway in E. coli by comparison of the carotenoid genes of Pantoea agglomerans and Pantoea ananatis. Appl. Microbiol. Biotechnol. 74, 131–139.

Zadeh, J.N., Steenberg, C.D., Bois, J.S., Wolfe, B.R., Pierce, M.B., Khan, A.R., Dirks, R.M., Pierce, N.A., 2011. NUPACK: analysis and design of nucleic acid systems. J. Comput. Chem. 32, 170–173.

Zamboni, N., Fischer, E., Sauer, U., 2005. FiatFlux—a software for metabolic flux analysis from 13C-glucose experiments. BMC Bioinf. 6, 209.