Seeing the wood for the trees: a forest of methods for optimisation and omic-network integration in metabolic modelling

Supreeta Vijayakumar*, Max Conway*, Pietro Lió and Claudio Angione

- Being downstream of gene expression, metabolism is being increasingly used as an indicator of the phenotypic outcome for drugs and therapies.
- We present an online resource and up-to-date classification of existing methods and tools for poly-omic modelling of metabolism in systems biology.
- We provide a hands-on tutorial for multi-objective optimisation of metabolic models in R.
- We finally discuss the implementation of multi-view machine-learning approaches in poly-omic data integration.

Supreeta Vijayakumar
Department of Computer Science and Information Systems, Teesside University, UK
Supreeta Vijayakumar is a PhD student at the Department of Computer Science and Information Systems, Teesside University, UK. Her research focuses on the integration of multi-omic data and machine learning algorithms for constraint-based modelling of microbial communities.
e-mail: s.vijayakumar@tees.ac.uk

Max Conway
Computer Laboratory, University of Cambridge, UK
Max Conway is a PhD student at the Computer Laboratory, University of Cambridge, UK. His research interests include machine learning to extract structure from metabolic and multi-omic models.
e-mail: max.conway@cl.cam.ac.uk

Pietro Lió
Computer Laboratory, University of Cambridge, UK
Pietro Lió is a Reader in Computational Biology at the Computer Laboratory, University of Cambridge. His affiliations also include the Cambridge Computational Biology Institute. His work spans Machine Learning and computational models for health Big Data, personalised medicine research, multi-scale/multi-omic/multi-physics modelling and data integration methods.
e-mail: pietro.lio@cl.cam.ac.uk

Claudio Angione
Department of Computer Science and Information Systems, Teesside University, UK
Claudio Angione is a Senior Lecturer in Computer Science at the Department of Computer Science and Information Systems, Teesside University, UK. He holds a PhD in Computer Science from the University of Cambridge, UK. His research interests include cancer metabolism, machine learning, systems biology and optimisation of genome-scale and poly-omic models.
e-mail: c.angione@tees.ac.uk

*These authors contributed equally to this work

Abstract Metabolic modelling has entered a mature phase with dozens of methods and software implementations available to the practitioner and the theoretician. It is not easy for a modeller to be able to see the wood (or the forest) for the trees. Driven by this analogy, we here present a "forest" of principal methods used for constraint-based modelling in systems biology. This provides a tree-based view of methods available to prospective modellers, also available in interactive version at http://modellingmetabolism.net, where it will be kept updated with new methods after the publication of the present manuscript. Our updated classification of existing methods and tools highlights the most promising in the different branches, with the aim to develop a vision of how existing methods could hybridise and become more complex. We then provide the first hands-on tutorial for multi-objective optimisation of metabolic models in R. We finally discuss the implementation of multi-view machine-learning approaches in poly-omic integration. Throughout this work, we demonstrate the optimisation of trade-offs between multiple metabolic objectives, with a focus on omic data integration through machine learning. We anticipate that the combination of a survey, a perspective on multi-view machine learning, and a step-by-step R tutorial should be of interest for both the beginner and the advanced user.

Introduction

Metabolism is the indispensable set of biochemical reactions in a cell that maintain its living state. Constraint-based reconstruction and analysis (COBRA) are a group of techniques that are commonly used for the mathematical and computational modelling of metabolic networks at the whole-genome scale. Genome-scale metabolic models are available in online repositories such as the Kyoto Encyclopedia of Genes and Genomes (KEGG) [1], the Biochemical Genetic and Genomic (BiGG) knowledge-base [2], the BioCyc collection of pathway/genome databases [3], MetaNetX [4] and the ModelSEED database [5]. Principally, the preparation of a genome-scale metabolic model involves the reconstruction of all metabolic reactions taking place in the organism supplemented with functional annotation of genes, metabolites and pathways. Depending on the quality of the reconstruction, processes of manual curation and gap-filling may also be required [6]. Predictions obtained from genome-scale metabolic models can be reconciled with *in-vivo* findings and used to identify current gaps in our knowledge of metabolism [7].

There are often inconsistencies between models and experimental data, such as when an outcome is falsely predicted by the model (false positive) or when an experimentally observed outcome is not predicted (false negative). Algorithms such as Grow Match [8], SMILEY [9] and optimal metabolic network identification (OMNI) [10] correct for such inconsistencies by suggesting adjustments for improving model accuracy. Reducing disparity between predicted and experimentally measured fluxes presents opportunities to devise new strategies for biological dis-

covery [11]. GapFind and GapFill are jointly designed optimisation procedures to identify 'problem' metabolites which cannot be produced or consumed in the network and then propose mechanisms to restore pathway connectivity for these metabolites [12]. fastGapFill is an extension of FASTCORE which incorporates flux and stoichiometric consistency into the gap-filling process [13]. Metabolic reconstruction via functional genomics (MIRAGE) conducts gap-filling by integrating with functional genomics data to estimate the probability of including each reaction from a universal database of gap-filling reactions in the reconstructed network [14]. This enables selection of the set of reactions whose addition is most likely to result in a fully functional model when flux analysis is repeated. Many models also integrate signalling and regulatory pathways with metabolic networks in order to add information regarding underlying mechanisms, consequently improving flux predictions [15].

Here, we present the foundations of constraint-based metabolic modelling as well as recent advances, in the form of a 'forest' of analytical tools and methods comprising algorithms and their software implementations. As such techniques are likely to expand and diversify over time, this schematic is also available in interactive version at http://modellingmetabolism.net, where it will be updated as newer methods are developed. We believe that classifying existing methods by their purpose or mode of implementation and defining their strengths and limitations will greatly facilitate the selection of methods for prospective modellers. In this regard, authors of new tools and methods are invited to contact us in order to include these in the interactive version of our figure.

In the following sections, we describe the main approaches currently used for constraint-based metabolic modelling, with the inclusion of many recent developments which we consider to be significant. These methods are divided into unbiased and biased approaches, the latter of which includes (i) a comprehensive review of flux balance analysis (FBA) and its specific variants, which apply different types of constraints for the prediction of metabolic fluxes (ii) regulatory methods, for which constraints are derived from external sources for designing context- or condition-specific metabolic models, and (iii) methods for the simulation of genetic perturbations and selection of the objective function. Following this, a detailed discussion of methods for performing multi-objective optimisation forms the basis of our tutorial for genetic design by multi-objective optimisation (GDMO) in R. Finally, we include a perspective that evaluates the potential use of multi-view machine learning techniques for the analysis of multi-omic metabolic models, which we regard as an important venture for the future of metabolic modelling.

Methods for constraint-based metabolic modelling

Figure 1 depicts the 'forest' of methods commonly used for constraint-based modelling of metabolic networks, following and updating the framework proposed by

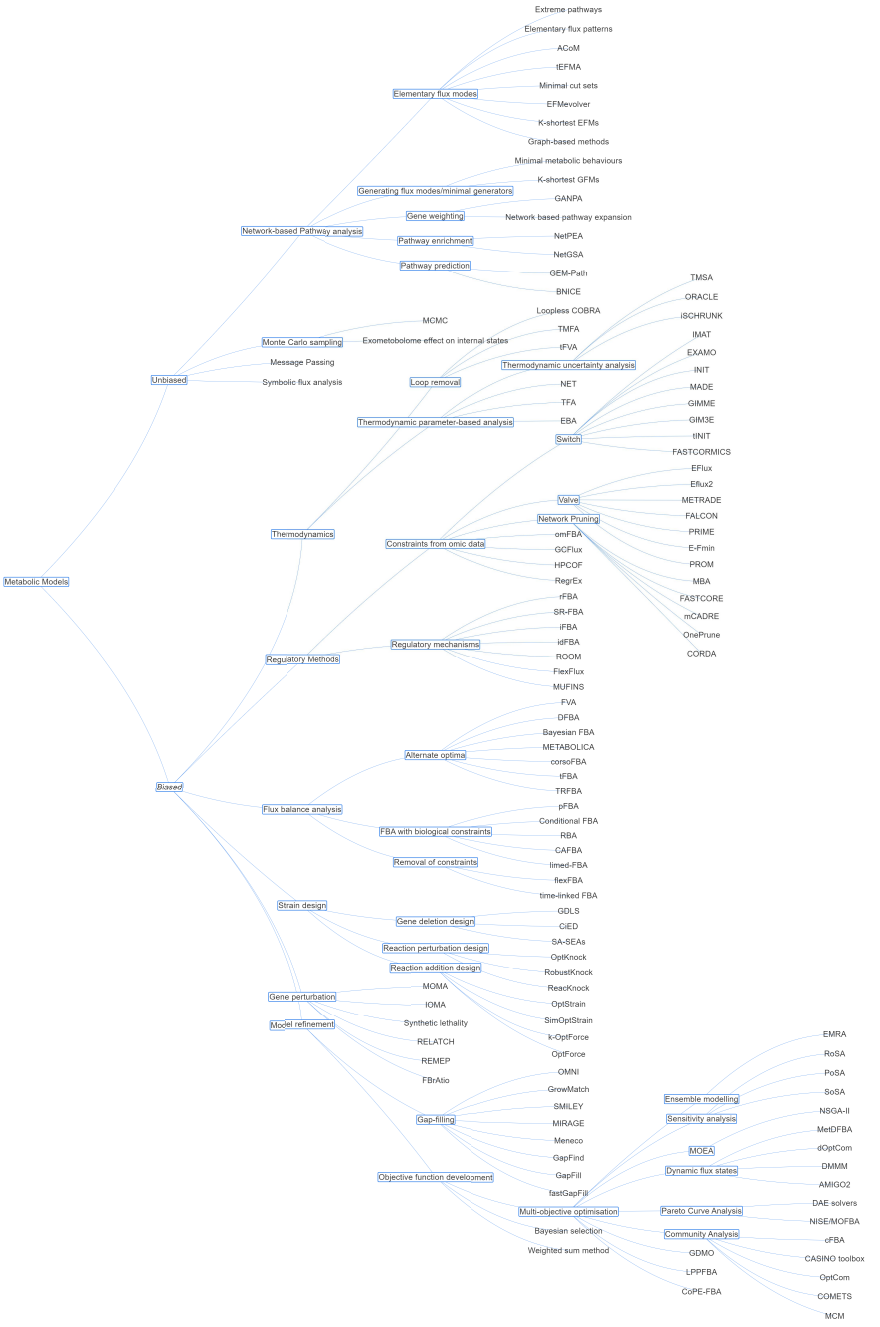

Fig. 1 A forest of methods based on constraint-based reconstruction and analysis (COBRA). Network-based pathway analysis describes the simplest configurations of metabolic pathways at steady state. Monte Carlo methods allow for uniform sampling of the solution space to compute the flux as a probability distribution for each pathway in the network. Thermodynamically infeasible fluxes may be eliminated using loop removal or thermodynamic parameter-based analysis. Model refinement may be carried out through the selection of single or multiple objectives or gap-filling techniques. Gene perturbation helps to establish the essentiality of genes and the most efficient pathways for the production of specific metabolites. Following this, strain design for metabolic engineering may involve strategies for gene deletion, reaction perturbation, or reaction addition. Alternate optimal solutions may be yielded by variations of flux balance analysis. Other variants may specify the addition of specific constraints or their removal. Additional constraints may be introduced through the inclusion of multi-omic data or gene regulatory mechanisms. Note that some methods may belong to multiple categories, but for clarity we classify each method according to its main contribution. An interactive version of this figure is maintained and updated at http://modellingmetabolism.net.

Lewis et al. [16]. Methodological approaches are broadly divided into biased and unbiased; the former necessitates the definition of an objective function by the network, whereas the latter relies on determining a subset of statistically analysable functional states whilst searching the entirety of the solution space [17].

Unbiased methods

Unbiased methods search the entirety of the solution space and find a subset of statistically analysable functional states without requiring the definition of an objective function. Network-based pathway analysis comprises a large family of unbiased methods assessing the main properties of biochemical pathways [18]. Gene Association Network-based Pathway Analysis (GANPA) improves upon this process by adding gene weighting to determine gene non-equivalence within pathways [19]. Similarly, a novel method was recently proposed for assessing the significance of pathways by constructing weighted gene-gene interaction networks for normal and cancerous tissue samples [20]. These interaction networks were subsequently used to expand pathways for each set of samples and compare their topologies. Approaches based on network-based pathway enrichment analysis aim to identify a greater number of gene interactions. For example, NetPEA utilises a protein-protein interaction (PPI) network combined with random walk to include information from high-throughput networks as well as known pathways [21]. A combination of network estimation with condition-specific omic data has been used to refine the NetGSA framework, thereby improving the ability to detect differential activity in pathways [22].

Using network-based pathway analysis, different methods may be used to calculate a set of routes through the reaction network and the corresponding kernel matrices which represent their stoichiometry. Elementary flux modes (EFMs) describe the minimal, non-decomposable set of pathways operating within a steady-state system; these are found by solving the steady-state condition following the iterative removal of single reactions until a valid flux distribution can no longer be calculated [23]. As this process often yields a combinatorial explosion of common functional motifs, a variation of the Agglomeration of Common Motifs (ACoM) method can be used to cluster these motifs, allowing for an overlap between classes [24]. Alternatively, a single EFM may be determined by solving an optimisation problem using EFMevolver [25], which can draw attention to significant EFMs. A method known as K-shortest EFMs enumerates EFMs in order of their number of reactions and has been applied to genome-scale networks [26]; the shortest pathways are of interest as they typically carry the highest flux and are easily manipulable.

A minimal generating set is the smallest set necessary to define the geometry of the flux space using a null-space algorithm, the elements of which are known as generating flux modes (GFMs) or minimal generators [27]. A variant of this method can be used to find specific subsets of GFMs in a process which does not result in a

combinatorial explosion, thus making them easier to compute [28]. Methods using minimal descriptions of the flux cone, such as minimal generators [29] and minimal metabolic behaviours [30] aim to reduce the dimensionality of the flux cone. A recent method prioritises the search for the shortest path between a pair of end nodes based on graph theory [31]. However, the validity of this approach has been questioned as reaction stoichiometry is overlooked [32]. tEFMA [33] removes thermodynamically-infeasible EFMs using network-embedded thermodynamic (NET) analysis [34]. The incorporation of thermodynamic constraints helps to select for physiologically significant EFMs, which become more difficult to detect as the size and complexity of networks increases. Identifying the largest thermodynamically consistent sets (LTCSs) in these EFMs can further characterise condition-specific metabolic capabilities in the thermodynamically-feasible regions of the flux cone [35].

Extreme pathways can be described as being a systemically independent subset of EFMs [18]. They are characterised by a set of convex basis vectors used to represent the edges of the steady-state solution space and consist of the minimum number of reactions needed to exist as a functional unit [36]. As opposed to many of methods described previously, extreme currents aim to increase dimensionality by describing non-decomposable EFMs situated both within and on the boundaries of the flux regions; these arise as a result of partitioning each reversible reaction into two irreversible reactions [37]. Minimal cut sets (MCSs) are another variant of EFMs that result in inactivity of the system with respect to the objective reaction if removed [38]. Therefore, they can be used to identify target genes and repress undesirable metabolic functions, whilst assessing the effect on the structure of the entire metabolic network. Elementary flux patterns (EFPs) define all potential elementary routes for steady-state fluxes as sets of indices, and can be mapped to EFMs to include factors such as pathway interdependencies, thus taking the entire network into account [39]. Frameworks which combine various computational approaches for synthetic pathway prediction, such as GEM-Path [40] and BNICE [41] are increasing in number as they provide the opportunity to calculate all possible paths and score them by efficiency [42].

Unbiased methods may also incorporate Monte Carlo sampling, message passing algorithms [43] or symbolic flux analysis [44]. The Markov chain Monte Carlo (MCMC) method can be used to uniformly sample metabolic networks from a genotype space, producing a sequence of viable genotypes (or reaction subsets) by performing a reaction swap between each genotype and its successor; if a swap results in a non-viable genotype, this sequence will remain at the previous genotype for that step and the process is repeated until a metabolic network with the correct number of reactions is reached [45]. A Monte Carlo based technique has also been used for uniform sampling of feasible steady states in an ellipsoid representing the solution space for a genome-scale metabolic model of *Escherichia coli* [46]. A revision of this method was proposed with rounding procedures to improve performance by eliminating ill-conditioning when sampling convex polytopes of steady states [47]. Of all omic data types, metabolomic data are said to give the closest indication of observed phenotypes [48]. Therefore, extracellular metabolomic measurements can

help to predict intracellular flux states by integrating these data into a constraint-based framework using a sampling-based network approach [49, 50]. The MetaboTools toolbox provides a workflow for integrating metabolomic data into multi-omic models and predicting metabolic phenotypes through analysing how metabolite uptake and secretion differ between conditions [48].

Biased methods

Biased methods rely on the definition of an objective function to solve the metabolic network and find its flux rates. For instance, standard flux balance analysis belongs to this class of methods.

Flux balance analysis and its variants

Among the biased methods, the most well-known technique is flux balance analysis (FBA), which uses the assignment of stoichiometric coefficients in a matrix to represent the metabolites involved in any given reaction in a metabolic network [51]. Through these coefficients, constraints can be imposed on the system to identify all potential flux distributions associated with a corresponding set of feasible phenotypic states. The aim of FBA is to locate a value (or set of values) in the solution space that best satisfies a given objective function. FBA uses linear programming to solve this objective function, indicating the extent to which each reaction in the network contributes to a phenotypic state.

If two objectives (flux rates or linear combinations thereof) were to be maximised, a multi-level linear problem would be formulated as follows:

$$\begin{array}{lll} \max & g^{\intercal}v & \\ \text{such that} & \max\ f^{\intercal}v & \\ & \text{such that} \quad Sv = 0 & \\ & v^{\min} \leq v \leq v^{\max} & \end{array} \tag{1}$$

where f and g are n-dimensional arrays of weights associated with the first and second objectives respectively, and indicate the contribution of the reaction fluxes v to each objective. $v^{\min}$ and $v^{\max}$ are vectors representing the lower and upper limits for the flux rates in v. A constraint in FBA postulates that the total amount of any metabolite being produced must be equal to the total amount of that metabolite consumed [52]. The most common objective function computed by FBA is the synthesis of biomass, which is commonly used to indicate cellular growth rate and predict product yields [53]. Fluxes can either be calculated under the steady state assumption or in a dynamic

state, where changes in specific concentrations and kinetics parameters have been recorded for each metabolite over time (e.g. for DFBA) [54]. Experimental validation of model predictions for DFBA are often obtained from ^{13}C metabolic flux analysis [55], which utilises isotopic-labelling of metabolic substrates to quantify intracellular fluxes. Additionally, methods such as dynamic multi-species metabolic modelling (DMMM) have been used to examine inter-species competition for metabolites in a microbial community [56].

Numerous modifications of FBA propose the application of various constraints to shrink the solution space for determining the precise flux state of the cell by calculating the optimal set of solutions for a given objective via linear programming. In most instances, constraints are defined by cell and reaction stoichiometry, fluxes through transport and metabolic reactions, upper and lower bounds for each flux, biomass composition and ATP requirements [57]. Upper and lower bounds can be estimated using flux variability analysis [58], which returns the maximum and minimum fluxes through each reaction whilst maintaining minimal biomass production [59].

Linear thermodynamic constraints can be applied in thermodynamic metabolic flux analysis (TMFA) (or thermodynamics-based flux balance analysis) and thermodynamic variability analysis to eliminate thermodynamically infeasible reactions or loops from pathways and gather information on feasible metabolite activity and Gibbs free energy changes [60, 61]. The removal of thermodynamically infeasible loops is necessary to prevent violating the loop law, which states that there is no net flux through balanced biochemical loops in networks at steady state [62]. Loopless COBRA methods solve a modified mixed-integer problem with the added constraint of no network fluxes containing loops; application of this constraint has been described for FBA, FVA and MCMC sampling [63]. FBA with thermodynamic constraints has also been described in energy-balance analysis (EBA) [64]. Fast flux variability analysis with thermodynamic constraints (tFVA) removes unbounded fluxes from biochemical loops arising from non-zero, steady-state fluxes involving internal reactions [65]. This is a faster implementation of FVA that does not require the specification of metabolite concentrations or additional experimental data, although these have been included in other variants of the method.

Parsimonious FBA (pFBA) identifies a subset of genes contributing to maximising the growth rate *in-silico*, therefore enabling maximisation of stoichiometric efficiency [66]. Another technique uses conditional dependencies present in the metabolic model as constraints for each flux, whereby each flux is constrained by the activity of the compound that facilitates it. This technique is known as conditional FBA and has proved to be effective for simulating phototrophic growth and diurnal dynamics in cyanobacteria [67, 68]. Resource balance analysis (RBA) uses growth rate limitation caused by distribution of proteins between cellular processes to constrain flux predictions [69]. Constrained allocation flux balance analysis (CAFBA) applies a genome-wide constraint on fluxes to observe proteome allocation between ribosomal, transport and biosynthetic proteins [70]. For this method, growth laws governing the synthesis of intracellular proteins are used to design parameters for predicting levels of protein expression and energy production. To improve the prediction of internal

fluxes, cost reduced sub-optimal FBA (corsoFBA) minimises protein and thermodynamic costs to simulate a sub-optimal state [71]. Linear metabolite dilution flux balance analysis (limed-FBA) forces dilution in metabolites associated with growth in active reactions, by adding a small dilution flux to block metabolic pathways without input fluxes [72].

Although more time-consuming than a linear programming approach, Bayesian flux estimation results in a probability density function, which is more stable and informative than a simple point estimate [73]. The METABOLICA statistical framework utilises a Bayesian approach to performing FBA. Metabolism is modelled in a multi-compartment macroscopic model with a stochastic extension of the stationary state and the Bayesian inference problem is solved by computing posterior probability densities using MCMC sampling [74].

Alternatively, standard constraints may be removed to construct new FBA methods to improve flux predictions for non-steady-state or non-wild-type cells. Relaxing the assumption of fixed reactant proportions for biomass production is the basis of flexible FBA (flexFBA), which can be coupled with relaxing the fixed ratio of byproduct to reactant (known as time-linked FBA) to observe transitions between steady states [75]. Combining these methods also enables the comparison of metabolite production between knockout mutants.

To further increase the informativity of constraint-based models, the integration of information from regulatory pathways and external multi-omic data is described in the following section.

Regulatory methods to generate context-specific metabolic models

Regulatory methods can be used to set constraints for FBA which incorporate regulatory networks, as well as introducing external omic data, which provides the opportunity to simulate metabolism under specific genetic or environmental conditions. Steady-state regulatory flux balance analysis (SR-FBA) is used to quantify the extent to which metabolic and transcriptional regulatory constraints affect the state of flux activity for various metabolic genes [76]. SR-FBA allows for improved characterisation of steady-state metabolic behaviour compared to regulatory FBA (rFBA), which chooses a single steady state per time interval from all possible solutions to find the flux distribution consistent with the regulatory state of each interval. Integrated FBA (iFBA) incorporates metabolic, regulatory and signalling pathways in the FBA model to enable thorough characterisation of dynamic-state metabolic behaviour [77]. Integrated dynamic flux balance analysis (idFBA) additionally couples fast and slow reactions to give quantitative time-variant flux predictions [78].

However, many of these approaches are limited by the Boolean logic formalism, which restricts the definition of gene activity to an on/off state. This disadvantage can be overcome by using conditional probabilities to represent gene states and

gene-transcription factor interactions when combining high throughput data with regulatory networks, as demonstrated by PROM [79]. This approach allows for a greater number of interactions between metabolic models and their respective transcriptional regulatory networks to be recorded as they are quantified automatically [80]. Other methods take a differential (rather than absolute) approach to gene expression analysis where gene expression levels classified as belonging to one of three states: over-expressed, unchanged or under-expressed [32]. FlexFlux jointly analyses multi-state regulatory networks and metabolic pathways, both of which contribute to calculation of flux [81]. Upon constructing an initial regulatory network, qualitative states evolving towards an 'attractor' set are converted into user-defined continuous intervals, thus allowing reactions to be constrained by multiple flux values, rather than one single value. The multi-formalism interaction network simulator (MUFINS) provides a platform for combining multiple kinetic models with signalling and regulatory networks, omic data integration algorithms and steady state FBA with linear inhibitor and activator constraints [82]. In this method, a quasi-steady state Petri net (QSSPN) is used to illustrate interactions between different networks [83].

Transcriptional-controlled FBA (tFBA) uses constraints between pairs of conditions based on gene expression data for the optimisation of FBA, considering both fold change and absolute change in expression to minimise noise [84]. Recently, a new method called transcriptional regulated flux balance analysis (TRFBA) has been introduced for incorporating expression data as well as transcriptional regulatory networks to simulate growth under various environmental and genetic perturbations [85]. TRFBA applies two unique linear constraints. Firstly, reaction rates are limited using a constant that sets expression levels equal to the upper bounds of reactions; secondly, the expression level of each gene is correlated with the expression of the regulating genes. One important advantage of TRFBA is the ability to improve predictions of growth without requiring detailed information about transcriptional regulators and their target genes.

For the integration of transcriptomic profiles with metabolic networks, gene expression measurements can be obtained from microarray and/or RNA sequencing data stored in public repositories such as the Gene Expression Omnibus (GEO) [86] or Array Express [87], in order to examine gene activity across various conditions. In addition to these primary archives, there are also databases with additional data processing and annotation [88] (such as information regarding gene regulation or differential expression under various conditions) e.g. Gene Expression Atlas [89], or the web servers Gene Chaser [90] and Profile Chaser [91], which query GEO. There are also many specialised databases providing functional genomic data relating to a particular disease, species or tissue-type, such as Oncomine (for cancer-specific microarrays) [92], the MGI Mouse Gene Expression Database (GXD) [93] or the Pancreatic Expression Database (PED) [94]. The generation of context-specific metabolic models may be divided into two main approaches: switch-based and valve-based methods.

Switch-based methods for omic integration. Switch-based algorithms remove inactive or lowly expressed genes by setting the corresponding reaction boundaries to

zero before FBA is performed [95]. For instance, the algorithm for gene inactivity moderated by metabolism and expression (GIMME) finds a flux distribution which optimises a given objective and avoids the use of so-called 'inactive' reactions below a certain transcription threshold [96, 97]. The main advantage of GIMME is that it can re-enable flux associated with false negative values in inactive reactions and record consistency between gene expression data and the predicted flux distribution for a given objective [98]. Gene inactivation moderated by metabolism, metabolomics and expression (GIM3E), is an extended version of GIMME which also incorporates metabolomic data in the form of turnover metabolites added as products to each reaction, along with a corresponding sink reaction [99]. This allows for the computation of turnover flux ranges for metabolites.

Tissue-specific gene and protein expression values can be integrated into genome-scale metabolic models accounting for different metabolic objectives at the cellular level [100], in order to extract information regarding the uptake and secretion of metabolites by specific tissue and cell-types. For this, tissue-specific variations in enzyme expression levels are used to inform the likelihood of enzymes supporting flux in their associated reactions by categorising gene-to-reaction mapping for each reaction in the model corresponding to the level of gene expression (i.e. high, moderate or low expression). Subsequently, fluxes corresponding to high gene expression are maximised and those corresponding to low gene expression are minimised when solving a mixed-integer linear program [101]. This process has been developed into the Integrative Metabolic Analysis Tool (iMAT) [102], which displays the most likely predicted metabolic fluxes corresponding to reactions in metabolic models. This tool enables the definition of a biological objective to be dependent on the requirements of each cell rather than the entire organism. An extension of iMAT known as the exploration of alternative metabolic optima (EXAMO) enables the design of condition-specific metabolic models for human tissues [103].

Similarly, the integrative network inference for tissues (INIT) algorithm uses tissue-specific information collected from the Human Protein Atlas to help incorporate transcriptomic and proteomic data into a genome-scale model and produce cell-type specific metabolic networks [104]. This data forms the input for a mixed-integer linear problem, which modifies the steady-state condition by setting a small positive net accumulation rate for internal metabolites [105]. Net productions of these metabolites are assigned positive weights, corresponding to arbitrary scores for the level of protein expression. An updated version of INIT known as the task-driven integrative network inference for tissues (tINIT) was devised to identify structural analogs to metabolites (so-called 'antimetabolites') and a core set of metabolic tasks to be included in the model [106]. tINIT prevents simultaneous flux in reversible reactions and allows the user to decide whether net production of all metabolites should be considered. The algorithm for metabolic adjustment by differential expression (MADE) compares the fold changes of gene expression values between conditions to intuitively predict the most consistent and statistically-significant metabolic adjustments [107]. The fold changes are expressed as a series of binary expression states, for which differences

between successive states most closely mirror corresponding differences in the mean expression levels.

Valve-based methods for omic integration. Unlike switch-based methods, valve-based algorithms reduce the activity of lowly expressed genes by adjusting the upper and lower bounds for their corresponding reactions. This is usually proportional to the normalised expression of the genes associated with those reactions before performing FBA [95]. Such methods include E-flux [108], E-flux2 [109], METRADE [110], FALCON [111] and PROM [79]. For valve-based methods, gene expression data is not discretised as in switch-based methods. Data from methods that treat gene expression as relative as opposed to absolute are more indicative of protein concentrations, as levels of transcription are more comparable across genes [112, 105]. In E-flux, flux boundaries are tightly constrained when gene expression is low but relaxed when gene expression is high [108]; transcript levels can be used to set an upper bound for the maximum production of enzymes and therefore constrain all reaction rates [97]. To improve this formulation, E-flux2 adds minimisation of the Euclidean norm of the measured flux vector, thus generating a unique solution [109].

Personalised reconstruction of metabolic models (PRIME) creates cell-specific models incorporating both transcriptomic and phenotypic data, and only modifies the bounds of a small set of reactions within a pre-defined range [113]. Expression data-guided flux minimisation (E-Fmin) is similar to GIMME in that it minimises a sum of fluxes where weights are a function of gene expression level; however, biomass production is forced to carry non-zero flux (i.e. metabolic activity is not threshold-dependent) and all reactions are thermodynamically-feasible due to flux minimisation [114]. FALCON is a novel algorithm that estimates enzyme abundances using gene-protein-reaction (GPR) rules in the model, thus improving the predictive capability of models integrated with expression data [111]. Within a multi-omic model, the metabolic and transcriptomics adaptation estimator (METRADE) constructs a Pareto front in order to identify the best trade-off when multiple objectives are simultaneously optimised [110]. Transcriptomic data comprising gene expression profiles and codon usage arrays can be mapped to a phenotypic objective space where each profile is associated with a condition [115]. In this way, the identification of optimal metabolic phenotypes is facilitated through the concurrent maximisation or minimisation of multiple metabolic markers and comparison of predicted flux rates between objectives.

Network pruning. In addition to switch- and valve- based integration methods, there are also pruning methods such as MBA [116], FASTCORE [117], mCADRE [118] and OnePrune [72], which only retain a core set of reactions in the metabolic model. FASTCORMICS is a faster adaptation of FASTCORE which facilitates data integration by pre-processing and produces multiple metabolic models [119]. Similarly, the cost optimisation reaction dependency assessment (CORDA) algorithm performs a four-step dependency assessment before calculating flux whilst minimising cost production i.e. utilising as many high confidence reactions as possible and minimising the involvement of absent reactions [120]. The cost of each reaction in the network is represented by the addition of a pseudo-metabolite as a product. CORDA is quicker

to implement than many other pruning methods owing to its use of FBA to calculate flux.

On the other hand, there are many methods do not fit into the aforementioned categories (switch/valve/network pruning) as they utilise more unconventional approaches for omic data integration. Similar to E-flux2, the regularised context-specific model extraction method (RegrEx) is based on principles of regularised least squares optimisation by minimising the squared Euclidean distance between fluxes and experimental data [121] to calculate fluxes which are independent of user-defined parameters. Instead of assigning expression measurements to individual genes or reactions, the GC-Flux algorithm splits GPR strings for each reaction into functional gene complexes to overcome the assumption of proteins catalysing more than one reaction at a time [122]. Another recent development is the use of the Huber penalty convex optimisation function (HPCOF) combined with flux minimisation, to achieve a more accurate prediction of fluxes which are closer to experimentally measured values [123]. This method introduces continuous gene expression values in the form of both constraints and target equations, without the need for definition of a biomass objective function or expression thresholds. A novel method known as omFBA [124] uses a phenotype-match algorithm to formulate the optimal objective function, i.e. the function that yields the most accurate estimations of the observed phenotypes. This objective can be simultaneously correlated with multiple omic data types via regression analysis to generate a omics-guided objective function, consequently resulting in a clearer correlation between genotype and phenotype and improved phenotypic predictions.

Genetic perturbation and objective function selection

Genetic perturbation is an important tool for establishing gene essentiality and maximising pathway efficiency. Deciding upon the number of gene perturbations to be performed depends on multiple factors. Choosing to perform single or pairwise gene perturbations one-by-one may fail to capture the essentiality and function of that gene as a result of genetic redundancy (i.e. there may be multiple genes encoding the same function). However, concurrently knocking out multiple genes can cause issues related to scaling unless coupled with e.g. Shapley value analysis, which assigns a contribution value to each gene knockout in the system [125]. Synthetic lethality can be described as the simultaneous inactivation of a set of non-essential genes resulting in the death of a cell or organism [126]. Knocking out multiple synthetic lethal pairs for genome-scale metabolic models can help in analysing the structural robustness of metabolic networks and identifying interdependencies among genes and reactions [127].

There are numerous algorithms for the detection and analysis of synthetic lethal pairs. Fast-SL is an algorithm capable of identifying higher order lethal reaction and

gene sets by taking GPR (gene-protein-reaction) associations into account and vastly reducing the search space before iterating through the remaining combinations of genes/reactions [128]. Minimal cut sets can also be regarded as synthetic lethals, which can be targeted for drug therapies as they constitute essential gene/reaction sets [129]. The data mining synthetic lethality identification pipeline (DAISY) statistically infers interactions between synthetic lethals using a combination of approaches: genomic survival of the fittest (SoF), shRNA-based functional examination and pairwise gene coexpression [130]. A method for identifying dosage lethality effects (IDLE) in genome-scale models of cancer metabolism exploits synthetic dosage lethality to simulate the pairwise knockout of non-essential enzymes via overexpression of the first gene and underexpression of the second [131].

Flux ratios can be applied as constraints for FBA using the flux balance analysis with flux ratios (FBrAtio) algorithm, which can be directly implemented into the stoichiometric matrix of genome-scale metabolic models [132, 133]. In FBrAtio, multiple enzymes compete for metabolic branch points in the network (known as critical nodes) [132], which specify how a substrate in the metabolite pool is distributed between competing reactions; this depends on factors relating to thermodynamics such as enzyme availability and downstream accumulation of reactive intermediates. The optimisation of flux ratios for a particular phenotype can be achieved through partial knockdown, overexpression or total knockout of enzyme-coding genes [132]. As opposed to complete knockouts, performing gene over-expression or partial knockdown may prove to be useful for targeted reduction of expression levels.

The minimisation of metabolic adjustment (MOMA) method relaxes the assumption of optimal growth flux for gene deletions by solving a quadratic problem to optimise distance minimisation in flux space [134]. This is because the minimal response to the perturbation is considered to be a more accurate estimate of the true flux state in the mutant [135]. Initially, the flux distribution for the mutant remains as close as possible to optimal flux for the wild-type and deviates to form a sub-optimal flux distribution between that of the wild-type and mutant. In this way, MOMA is able to predict phenotypic outcomes following knockouts more precisely than FBA.

Using a mechanistic model of reaction rates, integrative omics metabolic analysis (IOMA) also solves a quadratic problem to deliver kinetically-derived estimations of flux following genetic perturbation [136]. This is possible through integrating quantitative proteomic and metabolomic data into the model, which improves performance when compared to MOMA. Similarly, regulatory on/off minimisation (ROOM) minimises the number of significant flux changes following knockouts with respect to the wild type [137]. This is achieved through redirecting flux through alternative pathways following knockout.

Many gene perturbation experiments simulate knockouts under the assumption that there is no downstream effect on gene regulation [112]. In the RELATCH method, the principle of relative optimality is applied to predict how cells adapt to perturbations, by minimising relative flux patterns and latent pathway activation with respect to a reference flux distribution [138]. As strains adapt to their perturbed state,

they undergo regulatory and metabolic changes, represented by two parameters - one penalising latent pathway activation and another limiting enzyme contribution increases in active pathways. In varying environmental conditions, Bayesian factor modelling can be used to elucidate pathway cross-correlations and identify degrees of pathway activation [139]. Unconventionally, the REMEP method considers the impact of perturbations on metabolite as well as flux patterns [140]. This leads to improved flux predictions for knockout mutants as the structure of cellular regulation is represented more accurately.

Multi-objective optimisation of metabolic models

It can be difficult to define the single most important objective in a biological system as there are usually multiple conflicting cellular objectives in addition to the maximisation of biomass, which is often used as a proxy for growth. Methods such as Bayesian objective function discrimination can be used for selection of the most suitable objective function by using a probabilistic approach to compare multiple objectives [141]. Alternatively, there are methods utilising lexicographic ordering [142] or calculation of a weighted sum to scalarise multiple objectives [143]; however, it can be difficult to select weights that elicit a uniform distribution of Pareto solutions and find solutions in non-convex regions [144, 145, 146]. Thus, multi-objective optimisation arguably presents the most realistic representation of metabolic flux in biological systems by considering the contribution of a wide range of competing objectives. The rest of the section describes the main methods used to implement multi-objective optimisation in metabolic models.

Multi-objective optimisation can be used to resolve trade-offs between conflicting metabolic objectives through simulating a series of optimal, non-dominated vectors $f(x)$ in the multi-dimensional objective space. For such vectors, an improved solution does not exist for any given objective without sacrificing the performance of another [110]. This is known as a Pareto front and enables the simultaneous consideration of multiple conditions and constraints affecting each cellular objective [147]. For example, optimisation of the objective function r through maximisation or minimisation of the vector function $f(x)$ may be carried out respectively as follows:

$$\begin{aligned} f_i(x) > f_i(x^*),\ \forall\, i = 1, ..., r \\ f_i(x) < f_i(x^*),\ \forall\, i = 1, ..., r \end{aligned} \tag{2}$$

where x^* constitutes all non-dominated vectors present in the search space, for which there is no point x such that either of the above statements are satisfied (depending on whether a maximisation or minimisation is carried out). On the other hand, the non-linearity of metabolic networks means that there is often concavity or discontinuity present in Pareto fronts [148]. These issues may be resolved through the use of multi-objective evolutionary algorithms (MOEAs), through which it is possible

to obtain the entire set of Pareto-optimal solutions by running the algorithm only once [149]. Genetic design by multi-objective optimisation (GDMO) [150] employs MOEAs to find genetic manipulations (in the form of Pareto-optimal solutions) that simultaneously optimise multiple metabolic objectives.

The most widely used MOEA is NSGA-II [151], which conducts cross-comparisons between points in the objective space to establish whether a higher value exists for all objectives. This process is known as non-dominated sorting, as it involves categorising values in the distribution as either dominated or non-dominated. The non-dominated values are ordered into a front and the normalised distances between these points and their nearest neighbours are computed for each front. These are known as crowding distances and are necessary to preserve diversity (i.e. obtain a good spread of solutions), but may cause instability if two or more points share the same fitness values [152]. A sphere-excluding evolutionary algorithm (SEEA) has been proposed for maximising diversity in NSGA-II and preventing convergence at local optima [153]. Through comparing the non-dominated fronts, it is possible to establish the prioritisation of objectives for the production of specific metabolites.

In the interests of metabolic engineering, it is often useful to ascertain which knockouts would optimise the production of a specific metabolite. Commonly used for strain improvement in metabolic engineering, OptKnock is a computational method that identifies and subsequently removes metabolic reactions capable of uncoupling biomass maximisation from the production of a specific metabolite using a nested optimisation framework [154]. Cellular transport rates and secretion pathways can also be used to further constrain this multi-objective model [155]. RobustKnock is an extension of OptKnock that considers the role of competing pathways in diverting metabolic flux away from production of the desired metabolite, thus leading to suboptimal flux distributions [156]. Therefore, the removal of these competing pathways and improved knockout strategies result in more robust flux predictions. OptForce can be used to specify perturbations leading to targeted overproduction of a metabolite [157], with a variant known as k-OptForce which includes kinetic parameters if available (in the form of kinetic rate laws and metabolite concentrations) [158]. ReacKnock proposes an improved solution to previous methods defining strategies for deleting reactions, in that the Karush-Kuhn-Tucker (KKT) method is used to reformulate the problem for single-level optimisation [159]. As well as reaction deletions, OptStrain provides strategies for reaction addition through identifying non-native reactions in universal databases that are likely to improve product yields [160]. Furthermore, SimOptStrain considers gene deletions, addition of non-native reactions and gene-protein-reaction rules for optimising a given pathway [161]. Other strategies for gene deletion include genetic design through local search (GDLS) [162], the cipher of evolutionary design (CiED) [163], simulated annealing (SA) algorithms and set-based evolutionary algorithms (SEAs) [164].

Linear Physical Programming-Based Flux Balance Analysis (LPPFBA) aims to prioritise objectives and constraints for a given set of Pareto-optimal solutions, thus aiding the identification of conflicting objectives and regions of the solution space that contain feasible optimal fluxes [165]. Comprehensive Polyhedra Enumeration

Flux Balance Analysis (CoPE-FBA) indicates the topology of the sub-networks corresponding to optimal flux vectors in polyhedra with an emphasis on vertices of the solution space [166]. This process can be refined by dividing reversible reactions (termed linealities) into separate forward and backward reactions, thus simplifying the optimal solution space for optimisation of more objectives and yielding all non-decomposable flux routes [167].

Metabolic interactions between species in a microbial community have been simulated with respect to multiple objectives using algorithms such as OptCom, which provides a framework for the comparison of fitness trade-offs for individual species against that of the community [168]. Community flux balance analysis (cFBA) utilises non-linear multi-objective optimisation to build a more complete picture of metabolic fluxes by predicting biomass abundance as well as metabolite exchanges with the addition of community-specific constraints [169]. However, only flux distributions resulting in optimisation of the community growth rate are identified and the quality of flux predictions obtained using this method is heavily reliant on the quality of model reconstructions [170]. The community and systems-level interactive optimisation (CASINO) toolbox has been developed to conduct multi-level optimisation and phenotypic prediction for analysis of microbial interactions within a metabolic model of the human gut [171]. Here, biomass production for individual species as well as the microbial community as a whole is calculated, starting from a community matrix defining the topology of all reactions. The microbial community modeller (MCM) is another such tool where parametric fitting, sensitivity analysis and statistical evaluation are incorporated into models to assess the metabolic potential of each species in a community at the cellular level [172]. The computation of microbial ecosystems in time and space (COMETS) takes spatio-temporal dynamics into account by simulating time-dependent fluxes on a lattice containing information about the spatial distribution of microbial species and nutrients within a community; in addition to interactions, growth and uptake of substrates by different species can be simulated to examine how intracellular resource allocation is locally optimised by each species [173].

For multi-objective optimisation in dynamic flux states, changing intra- and extracellular concentrations of metabolites (and their corresponding gene expression values) can be observed over time using MetDFBA [174], which reduces the number of parameters for DFBA, thereby improving flux estimations. In this manner, different objective functions representing various phenotypes can be compared, and objectives that constrain fluxes corresponding to specific metabolites can be noted so that the feasible solution space can be further constrained. d-OptCom is a dynamic, multi-level extension of OptCom through which biomass accumulation and exchange of metabolites in microbial communities can be examined using dynamic mass balance equations and substrate uptake kinetics [175]. The AMIGO2 Toolbox reformulates optimal control problems as dynamic multi-objective optimisation problems, which can be solved using non-linear programming [176]. To circumvent the difficulties of acquiring time-course data to generate detailed kinetic models, ensemble modelling methods utilise steady state phenotypic data (such as flux changes caused by genetic

perturbations) to specify a set of models which represent the dynamics of a metabolic system [177]. Ensemble Modeling for Robustness Analysis (EMRA) considers how to maintain robustness in non-native pathways by building an ensemble of models linked to the same steady-state flux distribution. Subsequently, a continuation approach is used to alter kinetic parameters until a bifurcation point is found, where the steady state disappears [178].

Noninferior set estimation (NISE) has been used to divide the solution space and compute weights for multiple objectives in a multi-objective method for FBA (MOFBA). MOFBA can (i) generate Pareto curves demonstrating competing metabolic objectives and (ii) compute individual flux distributions for each Pareto optimal solution [179]. Large scale differential algebraic equation (DAE) solvers can be used to construct Pareto optimal curves for various perturbations such as heat shock, and observe where the nominal operating point for the wild type lies with respect to each curve [180]. There are variations of DAE systems that allow for sensitivity analysis to examine the rates of changes caused by small perturbations. Usually, sensitivity analysis examines the effect of changes in the reaction [181], pathway [150] and species [182] spaces of metabolic models [183]. The recently developed method for thermodynamics-based metabolite sensitivity analysis (TMSA) ranks metabolites on their ability to limit solutions to thermodynamically-consistent reactions and provides information about thermodynamic uncertainty in metabolic networks [184]. ORACLE (optimisation and risk analysis of complex living entities) evaluates quantitative uncertainty in kinetic models by sampling metabolite concentrations, and computing elasticity for enzyme states which represent the displacement of enzymes from thermodynamic equilibrium [185]. iSCHRUNK extends the ORACLE approach by including machine-learning classification algorithms to specify enzyme saturation levels and derive a more feasible population of kinetic models than ORACLE through achieving better characterisation of the kinetic parameters in the solution space [186].

In the following section, we present a hands-on tutorial for genetic design by multi-objective optimisation. Although several pipelines are available in Matlab and Python, this is the first tutorial available in R for multi-objective optimisation of FBA models.

Genetic design by multi-objective optimisation: an R tutorial

This tutorial on genetic design by multi-objective optimisation (GDMO) [150] shows how a multi-objective genetic optimisation algorithm (in this case, a modification of NSGA-II [151]) can be used to optimise the trade-off between multiple metabolic objectives. This is useful both as a tool for metabolic engineering (for example, when finding the favourable set of nutrients to optimise a microbial strain), and for quantifying the relationship between the objectives. The flowchart in Figure 2 provides a brief overview of the main stages of GDMO performed in R. Although we provide an explanation of the code, we suggest the following books for familiarisation

with the fundamentals of programming in R and evolutionary optimisation methods [187, 188, 189].

R, Python and Matlab all have good support for metabolic modelling. For a researcher approaching the field, the primary deciding factor is personal programming language preference. Performance is largely unrelated to language because the most time consuming step is optimising the linear optimisation problem, which is performed by an external linear programming toolkit. We have had particularly good results with the commercial solver Gurobi and the open source solver GLPK.

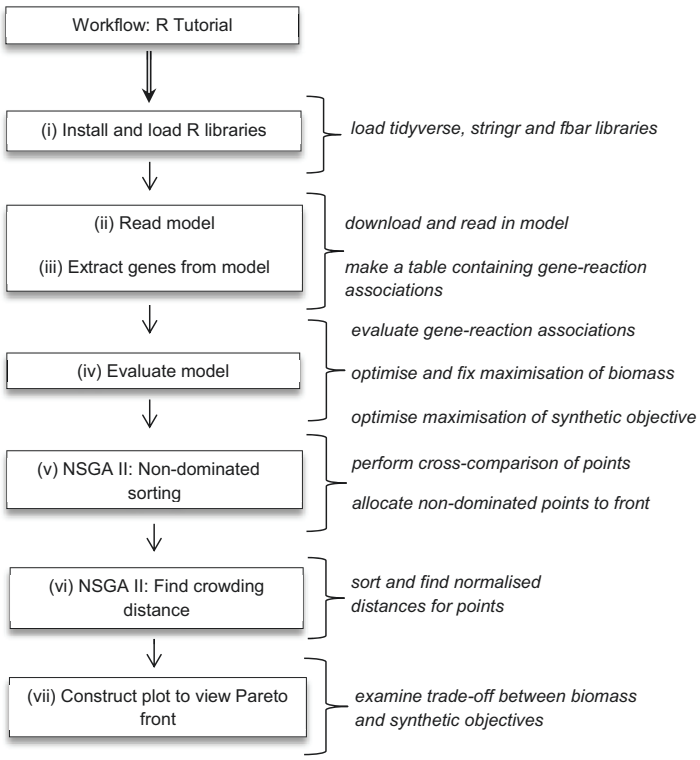

Fig. 2 A flowchart outlining the main stages of the R tutorial: (i) loading the requisite R libraries for analysis; (ii) reading the metabolic model into R; (iii) compiling a table that associates a list of genes extracted from the model with the reactions they are involved in; (iv) evaluating these associations by checking for the presence of genes in each iteration and performing FBA to obtain estimates of the biomass and synthetic objectives; (v) implementing a custom version of the NSGA-II algorithm, which conducts comparisons between points in the flux distribution to establish whether a higher value exists for all objectives, thus categorising these as dominated and non-dominated, the latter of which are ordered into a front; (vi) computing crowding distance i.e. the normalised distances between non-dominated points and their nearest neighbours in each front and dimension; (vii) viewing the Pareto front, through which it is possible to establish the prioritisation of objectives for the production of specific metabolites. The full code for this tutorial is provided in the Supplementary Material, and is downloadable from http://modellingmetabolism.net.

Loading and preparing a metabolic model are the first steps of any metabolic modelling procedure (see Supplementary Material). Assuming the initial steps have been carried out, we now describe the evaluation function that we use. In the context of multi-objective modelling, the evaluation function returns a number of values, each of which is used as an objective to be optimised. The evaluation function that we use here has four main stages:

1. The gene-reaction associations (`geneAssociation`) are evaluated in the context of which genes are present in this iteration (`genome`), to give an `activation` value, which is an estimate of reaction rate.
2. The `activation` value is used to alter the upper and lower bounds on reaction rate (`uppbnd` and `lowbnd`), to push reaction rates towards the rate estimates.
3. We conduct a round of FBA, optimising for maximum biomass.
4. We fix the biomass production value to its maximum, by altering the corresponding `uppbnd` and `lowbnd` to be near the `flux` (+/-1%).
5. With the biomass value fixed, we alter the objective coefficient (`obj_coef`) to target optimisation of the synthetic objective (acetate in the example).

The technique of fixing the biomass followed by maximising the synthetic objective is important because there could still be slack in the model after the first optimisation stage, which would allow for multiple possible synthetic objective values for a given biomass value and genome. This slack must be removed by a second optimisation round since we wish to have a correct and consistent estimate of the synthetic objective.

```
evaluation_function <- function(genome){

  res <- model %>%
    mutate(activation = fbar::gene_eval(expressions = geneAssociation,
                                        genes = names(genome),
                                        presences = genome
                                        ),
           activation = coalesce(activation, 1),
           uppbnd = pmin(uppbnd, 1000*activation+0.1),
           lowbnd = pmax(lowbnd, -1000*activation-0.1)) %>%
    fbar::find_fluxes_df(do_minimization = FALSE) %>%
    mutate(lowbnd = ifelse(abbreviation=='Biomass_Ecoli_core_w/GAM',
                           flux*0.99,
                           lowbnd),
           uppbnd = ifelse(abbreviation=='Biomass_Ecoli_core_w/GAM',
                           flux*1.01,
                           uppbnd),
           obj_coef = 1*(abbreviation=='EX_ac(e)')) %>%
    fbar::find_fluxes_df(do_minimization = FALSE)

  return(list(bm = filter(res, abbreviation=='Biomass_Ecoli_core_w/GAM')$flux,
              synth = filter(res, abbreviation=='EX_ac(e)')$flux))
}
```

Before proceeding with the genetic optimisation portion of the procedure, we need to describe two helper functions to our slightly modified form of NSGA-II, termed `non_dom_sort` and `crowding_distance`. Note that the following descriptions refer to the full code provided in the complete tutorial, available as Supplementary Material.

Non-domination sorting is the first stage of the selection procedure in NSGA-II. It sorts the points by multiple objectives to Pareto, or non-dominated fronts. These fronts are designed such that for every point in a front, there is no point in the same front or another front with a higher number such that the second point is better than the first in every objective (see Equation 2). This is calculated as follows:

1. We compare every point against every other point.
2. For each point (x), we see if there exists any second point (y) that has a higher value in all objectives. Where such a second point exists, we term the original point 'dominated'.
3. We find the set of points that have no dominating point, and term this the first non-dominated front. When two points are identical, they are both assigned to the same front.
4. We repeat this procedure, but ignore points in the first non-dominated front to find the second non-dominated front, and so on.

The second part of the NSGA-II evaluation procedure is finding the crowding distance. This is used to break ties between points in the same non-dominated front. For each front and for each dimension, this function sorts the points into order along the dimension, and finds the normalised distance between the proceeding point and succeeding point. These values are summed up across each dimension to find the value for the point.

The following code is the genetic loop of the algorithm. It is explained by code comments, but follows a normal pattern of evaluating, sorting, selecting from and mutating the population. The genetic algorithm used here is a modified version of NSGA-II [151], with a population of 200 individuals and carrying out 500 iterations. Inside the loop, the steps are as follows:

1. Evaluate all genomes: first, we use the evaluation function on each genome to find the resulting biomass and synthetic fluxes.
2. Round the results: this is a useful tool to help the NSGA-II procedure by regarding very similar results as identical, encouraging more variety in the results set.
3. Label the results: labelling is required so that we can identify them after non-dominance sorting.
4. Shuffle: shuffling the results is important because inevitably, some points will be completely identical, and we want to choose one at random in this case, rather than always pick the same result.

5. Find the non-dominated fronts: assign a front number to each point, such that points with a lower front number are strictly superior to those with a higher one.
6. Find the crowding distance: select for points with more variety.
7. Sort by front, breaking ties by crowding distance: there are normally multiple points in each front, so the continuous crowding distance value is required to choose between these.
8. Keep the best half of the population: using the label assigned earlier.
9. Sample parents from population: use random sampling with replacement to find which members of the population are used as a basis for new members.
10. Mutate parents to create offspring: add new members to the population by flipping 2% of the genes in the parents.
11. Combine the offspring and parent populations: build the new population and repeat.

```
start_genome <- set_names(rep_along(genes_in_model, TRUE), genes_in_model)
pop <- list(start_genome)

popsize = 200
generations = 500

for(i in 1:generations){
  results <- map_df(pop, evaluation_function) %>% # Evaluate all the genomes
    mutate(bm=signif(bm), synth=signif(synth)) %>% # Round results
    mutate(id = 1:n()) %>% # label the results
    sample_frac() %>% # Shuffle
    non_dom_sort() %>% # Find the non-dominated fronts
    crowding_distance() %>% # Find the crowding distances
    arrange(front, desc(crowding)) # Sort by front, breaking ties by crowding distance

  selected <- results %>%
    filter(row_number() <= popsize/2) %>% # Keep the best half of the population
    getElement('id')

  kept_pop <- pop[selected]

  altered_pop <- kept_pop %>%
    sample(popsize-length(selected), TRUE) %>% # Sample parents from population
    map(function(genome){
      xor(genome, runif(length(genome))>0.98) # Mutate parents to create offspring
      })

  pop <- unique(c(kept_pop, altered_pop)) # Combine the offspring and parent populations
}
```

Once we have a results set, we can construct a plot like Figure 3 to view the non-dominated fronts. We can see how the first front describes the trade-off between biomass and the synthetic objective, with the lines showing the dominated area (to the

bottom left). This shows that even with a small population and number of iterations we can see three high quality tradeoff points between the synthetic objective and biomass production, with biomass values of around 0.75, 0.63 and 0.38 h^{-1}, and corresponding synthetic values of 6, 11 and 14 mmol h^{-1} gDW^{-1}.

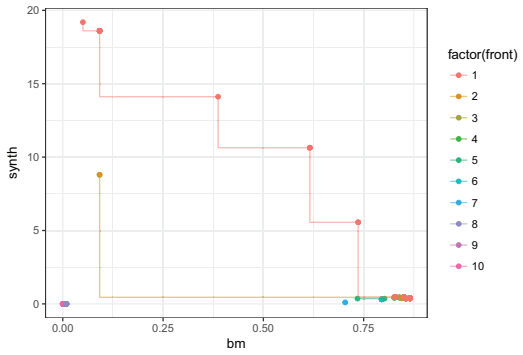

Fig. 3 Plot of a sample Pareto front showing the trade-off between the biomass (x-axis, h^{-1}) and the synthetic objective (y-axis, mmol h^{-1} gDW^{-1}). The area underlying each of the ten fronts of non-dominated points represents the number of metabolic configurations in the solution space that are supported when the objectives are maximised (or minimised).

Perspective: integration with multi-view machine learning approaches

Multi-view learning can be described as a sub-division of machine learning methods that aims to merge different aspects of a common problem in a single setting. It is based on principles of maximising the consensus between different viewpoints whilst offsetting the limitations of each view through complementation with the other views [190]. It is evident that this approach is highly applicable to the context of poly-omic data integration in genome scale metabolic models, owing to the interdependencies and correlations between all types of poly-omic data. At the very least, genomics, transcriptomics and proteomics are inextricably linked by the central dogma of molecular biology [191].

However, as omic types significantly differ in their scale and structure, the data is classed as heterogeneous and several normalisation measures would be required prior to mapping each omic as a layer in the metabolic model. Integration following the simultaneous analysis of multiple data types (known as meta-dimensional analysis) may be preceded by directly concatenating single sample matrices into one large matrix, transforming samples into intermediate graph or kernel matrices, or generating multiple models using different data types as training sets [192].

Data integration can be performed at an early, intermediate or late stage, depending on the nature of the data and the learning algorithm used. Early integration allows for the creation of a large pool of data before processing. Intermediate integration involves condensing each view into a similarity matrix with pairwise comparisons before a learning algorithm is applied. Late integration provides the opportunity to select the most suitable learning algorithm for each omic before merging data, comparing analyses between omics and linking patterns found in each omic [193]. As early integration increases data dimensionality rather than reducing it, intermediate or late integration would be preferred in this context to avoid introducing noise and decreasing performance prior to data transformation.

Methods such as clustering and multiple kernel learning have successfully been implemented in cancer sub-typing on the basis of shared molecular characteristics, and are ideal for classifying unsupervised data into groups to detect underlying associations where there is little information available [194, 195]. *K*-means is a traditional clustering algorithm that finds the number of clusters minimising the sum of the squared Euclidean distances between each observation and its respective cluster mean [196]. The algorithm starts by selecting k random points in the dataset (termed cluster centroids), which define the groups that the remaining data points are assigned to. The centroids are then moved to the averages computed in each group, and the process is repeated until distinctive clusters are formed. A number of flaws can be identified when this process is applied to multi-view learning. The primary concern is that of a lack of consideration of the importance of each individual view, as well as the differences between multiple views.

To address these issues, a bi-level variant of *k*-means clustering known as Tw-*k*-means was established, which added simultaneous weighting of both views and individual variables. This resulted in easier identification of the importance of variables and views, as well as in a decrease in the effect of low quality views and noise [197]. An alternative multi-view clustering approach known as iCluster [198] has been developed to model cancer subtypes from the Cancer Genome Atlas. In iCluster, cancer subtypes are considered as latent variables, which could be estimated by taking differences between views into account when partitioning multidimensional data into disparate groups. If a clustering algorithm is chosen for the partitioning of poly-omic data, it is important to note that the clusters of genes or metabolites may vary depending on the condition modelled. Integration by matrix factorisation (IMF) compiles clusters into matrices, which are factorised to assess the contribution of each separate cluster to each view, as well as the overall contribution of each view [199].

Multiple kernel learning transforms data structures into kernel matrices and optimises weight vectors that linearly combine these matrices to generate a unified kernel matrix. This facilitates the intermediate integration of data from different views, irrespective of the number of features utilised [195]. Another kernel-based technique known as similarity network fusion (SNF) [200] separately combines samples within each type of data to form individual networks. Such networks are iteratively integrated into a large, comprehensive network, mapped to the feature space in a non-linear fashion

and used to assess the amount of information of each data type in explaining any similarity observed between the samples. In a modified version of similarity network fusion [201], a bias layer was introduced between omic layers to account for the varying quality of metabolic reconstructions and therefore assign larger weights to omic layers that contributed more to the phenotype.

A support vector machine (SVM) is a supervised learning technique that, given a set of training examples, determines the optimal hyperplane to separate classes in the feature space whilst maximising the distance between samples of different classes [202]. Linear models such as SVMs or the least absolute shrinkage and selection operator (LASSO) can be treated as classification or regression problems respectively for performing feature concatenation with heterogeneous features, such as those found in poly-omic data [203].

Decision tree-based methods are highly intuitive and allow the analysis of both continuous and discrete features in the same model without the need for data normalisation. However, the high dimensionality of poly-omic datasets would make decision trees prone to noise and overfitting (especially if there are an insufficient number of features) [203]. This can be overcome by utilising ensemble learning methods such as random forest, which selects features at random as it constructs a decision tree. If a classification approach is taken, the most popular class is voted for following the generation of multiple trees; if the regression approach is taken, outputs from the multiple trees are averaged [204].

Feature selection utilises the minimum information necessary to classify key features and could prove useful in identifying the most significant trends in poly-omic datasets. Fortino et al. [205] have described a multivariate discovery process using "fuzzy patterns" to discretise and label gene expression data for the selection of the most relevant features. A random forest algorithm was then used to rank features in order of their usefulness, and to improve the stability and accuracy of data. A variation of this method has been used to implement feature selection in the integration of metabolomic, lipidomic and clinical data for the study of obesity and metabolic syndrome [206].

A number of Bayesian methods could also be applied to poly-omic data integration, such as Gaussian mixture models and Latent Dirichlet Allocation (LDA). Angione et al. [139] incorporated matrix factorisation into a Bayesian hierarchical model using Gaussian Markov Random Fields (GMRFs). This approach led to inferring cross-correlations between pathways in a metabolic network, and to prediction of pathway activation profiles as a result of bacterial responses to environmental conditions. Using a Bayesian method, predefined reaction-pathway memberships were used to model the prior distribution that was multiplied by the likelihood of observations to obtain the posterior distribution, which was subsequently used to infer model predictions. This approach serves to facilitate the observation of links between reactions, pathways and conditions, which can in turn help to interpret the consequences of metabolic flux variations and, consequently, the biological system as a whole. LDA has yet to be applied to poly-omic integration but has potential

because of its efficacy in organising unsupervised data from a mixture of clusters [207].

Conclusion

In this work, we have surveyed the principal methods available for constraint-based modelling and omic integration. We have presented these in the form of a 'forest', also available in interactive version at http://modellingmetabolism.net where it will be updated periodically. As an up-to-date classification of available methods, we believe that this will prove to be a useful resource for prospective modellers. We have also provided the first tutorial in R for multi-objective optimisation of metabolic models.

We envisage that a late integration approach can be used to test the suitability of various multi-view learning algorithms for each omic dataset used in the integration process, depending on the structure of the data. For example, a clustering algorithm may be chosen for mapping microarray or RNA sequencing data onto a metabolic model. Likewise, a SVM may be chosen for mapping protein data or post-translational modifications. Finally, feature selection may be employed to identify the most distinctive features across both of these omic layers, as a whole.

In the near future, the integration of multi-view machine learning in metabolic modelling is likely to grow in parallel with the rapid advancement of high-throughput omic technologies. With the increase in the number of omic layers, existing algorithms will continue to be extended. However, we believe a one-size-fits-all approach is not the most effective way of tackling a multi-omic genome-scale problem. Given the intrinsic differences between omic layers, we envisage specific algorithms designed only for the analysis of each layer. When required, a global prediction can then be achieved through methods for aggregation of layers, such as those successfully employed in multilayer network theory.

References

1. Kanehisa M, Goto S. KEGG: kyoto encyclopedia of genes and genomes. Nucleic acids research. 2000;28(1):27–30.
2. Schellenberger J, Park JO, Conrad TM, et al. BiGG: a Biochemical Genetic and Genomic knowledgebase of large scale metabolic reconstructions. BMC Bioinformatics. 2010;11(1):1–10.
3. Karp PD, Ouzounis CA, Moore-Kochlacs C, et al. Expansion of the BioCyc collection of pathway/genome databases to 160 genomes. Nucleic acids research. 2005;33(19):6083–6089.

4. Moretti S, Martin O, Van Du Tran T, et al. MetaNetX/MNXref–reconciliation of metabolites and biochemical reactions to bring together genome-scale metabolic networks. Nucleic acids research. 2016;44(D1):D523–D526.
5. Henry CS, DeJongh M, Best AA, et al. High-throughput generation, optimization and analysis of genome-scale metabolic models. Nature biotechnology. 2010;28(9):977–982.
6. Prigent S, Frioux C, Dittami SM, et al. Meneco, a Topology-Based Gap-Filling Tool Applicable to Degraded Genome-Wide Metabolic Networks. PLoS Computational Biology. 2017;13(1):e1005276.
7. Mienda BS. Genome-scale metabolic models as platforms for strain design and biological discovery. Journal of Biomolecular Structure and Dynamics. 2016;just-accepted(just-accepted):1–23.
8. Kumar VS, Maranas CD. GrowMatch: an automated method for reconciling in silico/in vivo growth predictions. PLoS Comput Biol. 2009;5(3):e1000308.
9. Reed JL, Patel TR, Chen KH, et al. Systems approach to refining genome annotation. Proceedings of the National Academy of Sciences. 2006;103(46):17480–17484.
10. Herrgård MJ, Fong SS, Palsson BØ. Identification of genome-scale metabolic network models using experimentally measured flux profiles. PLoS Comput Biol. 2006;2(7):e72.
11. OâĂŹBrien EJ, Monk JM, Palsson BO. Using genome-scale models to predict biological capabilities. Cell. 2015;161(5):971–987.
12. Kumar VS, Dasika MS, Maranas CD. Optimization based automated curation of metabolic reconstructions. BMC bioinformatics. 2007;8(1):212.
13. Thiele I, Vlassis N, Fleming RM. fastGapFill: efficient gap filling in metabolic networks. Bioinformatics. 2014;30(17):2529–2531.
14. Vitkin E, Shlomi T. MIRAGE: a functional genomics-based approach for metabolic network model reconstruction and its application to cyanobacteria networks. Genome biology. 2012;13(11):1.
15. Palsson BØ. Systems Biology: Constraint-Based Reconstruction and Analysis. Cambridge University Press; 2015.
16. Lewis NE, Nagarajan H, Palsson BØ. Constraining the metabolic genotype–phenotype relationship using a phylogeny of in silico methods. Nature Reviews Microbiology. 2012;10(4):291–305.
17. Lee SY, et al. Systems biology and biotechnology of Escherichia coli. Springer; 2009.
18. Papin JA, Stelling J, Price ND, et al. Comparison of network-based pathway analysis methods. Trends in biotechnology. 2004;22(8):400–405.
19. Fang Z, Tian W, Ji H. A network-based gene-weighting approach for pathway analysis. Cell research. 2012;22(3):565–580.
20. Zhang Q, Li J, Xie H, et al. A network-based pathway-expanding approach for pathway analysis. BMC Bioinformatics. 2016;17(17):231.
21. Liu L, Ruan J. Network-based pathway enrichment analysis. In: Bioinformatics and Biomedicine (BIBM), 2013 IEEE International Conference on. IEEE; 2013. p. 218–221.
22. Ma J, Shojaie A, Michailidis G. Network-based pathway enrichment analysis with incomplete network information. Bioinformatics. 2016;32(20):3165–3174.
23. Zanghellini J, Ruckerbauer DE, Hanscho M, et al. Elementary flux modes in a nutshell: properties, calculation and applications. Biotechnology journal. 2013;8(9):1009–1016.
24. Pérès S, Vallée F, Beurton-Aimar M, et al. ACoM: a classification method for elementary flux modes based on motif finding. BioSystems. 2011;103(3):410–419.
25. Kaleta C, De Figueiredo LF, Behre J, et al. EFMEvolver: Computing elementary flux modes in genome-scale metabolic networks. Lect Notes Inform. 2009;p. 179–89.
26. De Figueiredo LF, Podhorski A, Rubio A, et al. Computing the shortest elementary flux modes in genome-scale metabolic networks. Bioinformatics. 2009;25(23):3158–3165.
27. Wagner C, Urbanczik R. The geometry of the flux cone of a metabolic network. Biophysical journal. 2005;89(6):3837–3845.
28. Rezola A, de Figueiredo LF, Brock M, et al. Exploring metabolic pathways in genome-scale networks via generating flux modes. Bioinformatics. 2011;27(4):534–540.

29. Urbanczik R, Wagner C. An improved algorithm for stoichiometric network analysis: theory and applications. Bioinformatics. 2005;21(7):1203–1210.
30. Larhlimi A, Bockmayr A. A new constraint-based description of the steady-state flux cone of metabolic networks. Discrete Applied Mathematics. 2009;157(10):2257–2266.
31. Hidalgo JF, Guil F, García JM. A new approach to obtaining EFMs using graph methods based on the shortest path between end nodes. Genomics and Computational Biology. 2016;2(1):30.
32. Rezola A, Pey J, Tobalina L, et al. Advances in network-based metabolic pathway analysis and gene expression data integration. Briefings in bioinformatics. 2014;p. bbu009.
33. Gerstl MP, Ruckerbauer DE, Mattanovich D, et al. Metabolomics integrated elementary flux mode analysis in large metabolic networks. Scientific reports. 2015;5:8930.
34. Kümmel A, Panke S, Heinemann M. Putative regulatory sites unraveled by network-embedded thermodynamic analysis of metabolome data. Molecular systems biology. 2006;2(1).
35. Gerstl MP, Jungreuthmayer C, Müller S, et al. Which sets of elementary flux modes form thermodynamically feasible flux distributions? FEBS Journal. 2016;283(9):1782–1794.
36. Schilling CH, Letscher D, Palsson BØ. Theory for the systemic definition of metabolic pathways and their use in interpreting metabolic function from a pathway-oriented perspective. Journal of theoretical biology. 2000;203(3):229–248.
37. Clarke BL. Stoichiometric network analysis. Cell Biochemistry and Biophysics. 1988;12(1):237–253.
38. Clark ST, Verwoerd WS. Minimal cut sets and the use of failure modes in metabolic networks. Metabolites. 2012;2(3):567–595.
39. Kaleta C, de Figueiredo LF, Schuster S. Can the whole be less than the sum of its parts? Pathway analysis in genome-scale metabolic networks using elementary flux patterns. Genome Research. 2009;19(10):1872–1883.
40. Campodonico MA, Andrews BA, Asenjo JA, et al. Generation of an atlas for commodity chemical production in Escherichia coli and a novel pathway prediction algorithm, GEM-Path. Metabolic engineering. 2014;25:140–158.
41. Hatzimanikatis V, Li C, Ionita JA, et al. Exploring the diversity of complex metabolic networks. Bioinformatics. 2005;21(8):1603–1609.
42. Medema MH, Van Raaphorst R, Takano E, et al. Computational tools for the synthetic design of biochemical pathways. Nature Reviews Microbiology. 2012;10(3):191–202.
43. Braunstein A, Mulet R, Pagnani A. Estimating the size of the solution space of metabolic networks. BMC bioinformatics. 2008;9(1):240.
44. Schryer DW, Vendelin M, Peterson P. Symbolic flux analysis for genome-scale metabolic networks. BMC Systems Biology. 2011;5(1):1–13.
45. Samal A, Matias Rodrigues JF, Jost J, et al. Genotype networks in metabolic reaction spaces. BMC Systems Biology. 2010;4(1):1–21.
46. De Martino D, Parisi V. Montecarlo uniform sampling of highdimensional convex polytopes: reducing the condition number with applications in metabolic network analysis. arXiv preprint. arXiv preprint arXiv:13125228. 2013;.
47. De Martino D, Mori M, Parisi V. Uniform sampling of steady states in metabolic networks: heterogeneous scales and rounding. PloS one. 2015;10(4):e0122670.
48. Aurich MK, Fleming RM, Thiele I. MetaboTools: A comprehensive toolbox for analysis of genome-scale metabolic models. Frontiers in Physiology. 2016;7.
49. Mo ML, Palsson BØ, Herrgård MJ. Connecting extracellular metabolomic measurements to intracellular flux states in yeast. BMC Systems Biology. 2009;3(1):1–17.
50. Aurich MK, Paglia G, Rolfsson Ó, et al. Prediction of intracellular metabolic states from extracellular metabolomic data. Metabolomics. 2015;11(3):603–619.
51. Orth JD, Thiele I, Palsson BØ. What is flux balance analysis? Nature Biotechnology. 2010;28(3):245–248.
52. Zieliński ŁP, Smith AC, Smith AG, et al. Metabolic flexibility of mitochondrial respiratory chain disorders predicted by computer modelling. Mitochondrion. 2016;31:45–55.
53. Feist AM, Palsson BØ. The biomass objective function. Current opinion in microbiology. 2010;13(3):344–349.

54. Mahadevan R, Edwards JS, Doyle FJ. Dynamic flux balance analysis of diauxic growth in Escherichia coli. Biophysical journal. 2002;83(3):1331–1340.
55. Wiechert W. 13 C metabolic flux analysis. Metabolic engineering. 2001;3(3):195–206.
56. Zhuang K, Izallalen M, Mouser P, et al. Genome-scale dynamic modeling of the competition between Rhodoferax and Geobacter in anoxic subsurface environments. The ISME journal. 2011;5(2):305–316.
57. Reed JL. Shrinking the metabolic solution space using experimental datasets. PLoS Comput Biol. 2012;8(8):e1002662.
58. Burgard AP, Vaidyaraman S, Maranas CD. Minimal reaction sets for Escherichia coli metabolism under different growth requirements and uptake environments. Biotechnology progress. 2001;17(5):791–797.
59. Mahadevan R, Schilling C. The effects of alternate optimal solutions in constraint-based genome-scale metabolic models. Metabolic engineering. 2003;5(4):264–276.
60. Henry CS, Broadbelt LJ, Hatzimanikatis V. Thermodynamics-based metabolic flux analysis. Biophysical journal. 2007;92(5):1792–1805.
61. Ataman M, Hatzimanikatis V. Heading in the right direction: thermodynamics-based network analysis and pathway engineering. Current Opinion in Biotechnology. 2015;36:176–182.
62. Price ND, Famili I, Beard DA, et al. Extreme pathways and Kirchhoff's second law. Biophysical journal. 2002;83(5):2879.
63. Schellenberger J, Lewis NE, Palsson BØ. Elimination of thermodynamically infeasible loops in steady-state metabolic models. Biophysical journal. 2011;100(3):544–553.
64. Beard DA, Liang Sd, Qian H. Energy balance for analysis of complex metabolic networks. Biophysical journal. 2002;83(1):79–86.
65. Müller AC, Bockmayr A. Fast thermodynamically constrained flux variability analysis. Bioinformatics. 2013;p. btt059.
66. Lewis NE, Hixson KK, Conrad TM, et al. Omic data from evolved E. coli are consistent with computed optimal growth from genome-scale models. Molecular systems biology. 2010;6(1):390.
67. Rügen M, Bockmayr A, Steuer R. Elucidating temporal resource allocation and diurnal dynamics in phototrophic metabolism using conditional FBA. Scientific Reports. 2015;5.
68. Reimers AM, Knoop H, Bockmayr A, et al. Evaluating the stoichiometric and energetic constraints of cyanobacterial diurnal growth. arXiv preprint arXiv:161006859. 2016;.
69. Goelzer A, Fromion V. Bacterial growth rate reflects a bottleneck in resource allocation. Biochimica et Biophysica Acta (BBA)-General Subjects. 2011;1810(10):978–988.
70. Mori M, Hwa T, Martin OC, et al. Constrained allocation flux balance analysis. PLoS Comput Biol. 2016;12(6):e1004913.
71. Schultz A, Qutub AA. Predicting internal cell fluxes at sub-optimal growth. BMC systems biology. 2015;9(1):18.
72. Dreyfuss JM, Zucker JD, Hood HM, et al. Reconstruction and validation of a genome-scale metabolic model for the filamentous fungus Neurospora crassa using FARM. PLoS Comput Biol. 2013;9(7):e1003126.
73. Heino J, Tunyan K, Calvetti D, et al. Bayesian flux balance analysis applied to a skeletal muscle metabolic model. Journal of theoretical biology. 2007;248(1):91–110.
74. Heino J, Calvetti D, Somersalo E. Metabolica: a statistical research tool for analyzing metabolic networks. Computer methods and programs in biomedicine. 2010;97(2):151–167.
75. Birch EW, Udell M, Covert MW. Incorporation of flexible objectives and time-linked simulation with flux balance analysis. Journal of theoretical biology. 2014;345:12–21.
76. Shlomi T, Eisenberg Y, Sharan R, et al. A genome-scale computational study of the interplay between transcriptional regulation and metabolism. Molecular systems biology. 2007;3(1):101.
77. Covert MW, Xiao N, Chen TJ, et al. Integrating metabolic, transcriptional regulatory and signal transduction models in Escherichia coli. Bioinformatics. 2008;24(18):2044–2050.
78. Lee JM, Gianchandani EP, Eddy JA, et al. Dynamic analysis of integrated signaling, metabolic, and regulatory networks. PLoS Comput Biol. 2008;4(5):e1000086.

79. Chandrasekaran S, Price ND. Probabilistic integrative modeling of genome-scale metabolic and regulatory networks in Escherichia coli and Mycobacterium tuberculosis. Proceedings of the National Academy of Sciences. 2010;107(41):17845–17850.
80. Chandrasekaran S, Price ND. Metabolic constraint-based refinement of transcriptional regulatory networks. PLoS Comput Biol. 2013;9(12):e1003370.
81. Marmiesse L, Peyraud R, Cottret L. FlexFlux: combining metabolic flux and regulatory network analyses. BMC Systems Biology. 2015;9(1):93.
82. Wu H, Von Kamp A, Leoncikas V, et al. MUFINS: multi-formalism interaction network simulator. npj Systems Biology and Applications. 2016;2:16032.
83. Fisher CP, Plant NJ, Moore JB, et al. QSSPN: dynamic simulation of molecular interaction networks describing gene regulation, signalling and whole-cell metabolism in human cells. Bioinformatics. 2013;29(24):3181–3190.
84. van Berlo RJ, de Ridder D, Daran JM, et al. Predicting metabolic fluxes using gene expression differences as constraints. IEEE/ACM Transactions on Computational Biology and Bioinformatics. 2011;8(1):206–216.
85. Motamedian E, Mohammadi M, Shojaosadati SA, et al. TRFBA: an algorithm to integrate genome-scale metabolic and transcriptional regulatory networks with incorporation of expression data. Bioinformatics. 2017;p. btw772.
86. Barrett T, Wilhite SE, Ledoux P, et al. NCBI GEO: archive for functional genomics data setsâĂŤupdate. Nucleic acids research. 2013;41(D1):D991–D995.
87. Kolesnikov N, Hastings E, Keays M, et al. ArrayExpress updateâĂŤsimplifying data submissions. Nucleic acids research. 2014;p. gku1057.
88. Rung J, Brazma A. Reuse of public genome-wide gene expression data. Nature Reviews Genetics. 2013;14(2):89–99.
89. Petryszak R, Keays M, Tang YA, et al. Expression Atlas updateâĂŤan integrated database of gene and protein expression in humans, animals and plants. Nucleic acids research. 2015;p. gkv1045.
90. Chen R, Mallelwar R, Thosar A, et al. GeneChaser: Identifying all biological and clinical conditions in which genes of interest are differentially expressed. BMC Bioinformatics. 2008;9(1):548.
91. Engreitz JM, Chen R, Morgan AA, et al. ProfileChaser: searching microarray repositories based on genome-wide patterns of differential expression. Bioinformatics. 2011;27(23):3317–3318.
92. Rhodes DR, Yu J, Shanker K, et al. ONCOMINE: a cancer microarray database and integrated data-mining platform. Neoplasia. 2004;6(1):1–6.
93. Finger JH, Smith CM, Hayamizu TF, et al. The mouse Gene Expression Database (GXD): 2017 update. Nucleic Acids Research. 2017;45(D1):D730–D736.
94. Ullah AZD, Cutts RJ, Ghetia M, et al. The pancreatic expression database: recent extensions and updates. Nucleic Acids Research. 2014;42(D1):D944–D949.
95. Salehzadeh-Yazdi A, Asgari Y, Saboury AA, et al. Computational analysis of reciprocal association of metabolism and epigenetics in the budding yeast: a genome-scale metabolic model (GSMM) approach. PloS one. 2014;9(11):e111686.
96. Vivek-Ananth R, Samal A. Advances in the integration of transcriptional regulatory information into genome-scale metabolic models. Biosystems. 2016;147:1–10.
97. Kim MK, Lun DS. Methods for integration of transcriptomic data in genome-scale metabolic models. Computational and structural biotechnology journal. 2014;11(18):59–65.
98. Becker SA, Palsson BO. Context-specific metabolic networks are consistent with experiments. PLoS Comput Biol. 2008;4(5):e1000082.
99. Schmidt BJ, Ebrahim A, Metz TO, et al. GIM3E: condition-specific models of cellular metabolism developed from metabolomics and expression data. Bioinformatics. 2013;29(22):2900–2908.
100. Shlomi T, Cabili MN, Herrgård MJ, et al. Network-based prediction of human tissue-specific metabolism. Nature biotechnology. 2008;26(9):1003–1010.
101. Blazier AS, Papin JA. Integration of expression data in genome-scale metabolic network reconstructions. Frontiers in physiology. 2012;3:299.

102. Zur H, Ruppin E, Shlomi T. iMAT: an integrative metabolic analysis tool. Bioinformatics. 2010;26(24):3140–3142.
103. Rossell S, Huynen MA, Notebaart RA. Inferring metabolic states in uncharacterized environments using gene-expression measurements. PLoS Comput Biol. 2013;9(3):e1002988.
104. Agren R, Bordel S, Mardinoglu A, et al. Reconstruction of genome-scale active metabolic networks for 69 human cell types and 16 cancer types using INIT. PLoS Comput Biol. 2012;8(5):e1002518.
105. Machado D, Herrgård M. Systematic evaluation of methods for integration of transcriptomic data into constraint-based models of metabolism. PLoS Comput Biol. 2014;10(4):e1003580.
106. Agren R, Mardinoglu A, Asplund A, et al. Identification of anticancer drugs for hepatocellular carcinoma through personalized genome-scale metabolic modeling. Molecular systems biology. 2014;10(3):721.
107. Jensen PA, Papin JA. Functional integration of a metabolic network model and expression data without arbitrary thresholding. Bioinformatics. 2011;27(4):541–547.
108. Colijn C, Brandes A, Zucker J, et al. Interpreting expression data with metabolic flux models: predicting Mycobacterium tuberculosis mycolic acid production. PLoS Comput Biol. 2009;5(8):e1000489.
109. Kim MK, Lane A, Kelley JJ, et al. E-Flux2 and SPOT: validated methods for inferring intracellular metabolic flux distributions from transcriptomic data. PLoS One. 2016;11(6):e0157101.
110. Angione C, Lió P. Predictive analytics of environmental adaptability in multi-omic network models. Scientific Reports. 2015;5:15147.
111. Barker BE, Sadagopan N, Wang Y, et al. A robust and efficient method for estimating enzyme complex abundance and metabolic flux from expression data. Computational biology and chemistry. 2015;59:98–112.
112. Lee D, Smallbone K, Dunn WB, et al. Improving metabolic flux predictions using absolute gene expression data. BMC systems biology. 2012;6(1):73.
113. Yizhak K, Gaude E, Le Dévédec S, et al. Phenotype-based cell-specific metabolic modeling reveals metabolic liabilities of cancer. Elife. 2014;3:e03641.
114. Song HS, Reifman J, Wallqvist A. Prediction of metabolic flux distribution from gene expression data based on the flux minimization principle. PloS one. 2014;9(11):e112524.
115. Kashaf SS, Angione C, Lió P. Making life difficult for Clostridium difficile: augmenting the pathogen's metabolic model with transcriptomic and codon usage data for better therapeutic target characterization. BMC Systems Biology. 2017;11(1):25.
116. Jerby L, Shlomi T, Ruppin E. Computational reconstruction of tissue-specific metabolic models: application to human liver metabolism. Molecular systems biology. 2010;6(1):401.
117. Vlassis N, Pacheco MP, Sauter T. Fast reconstruction of compact context-specific metabolic network models. PLoS Comput Biol. 2014;10(1):e1003424.
118. Wang Y, Eddy JA, Price ND. Reconstruction of genome-scale metabolic models for 126 human tissues using mCADRE. BMC systems biology. 2012;6(1):153.
119. Pacheco MP, John E, Kaoma T, et al. Integrated metabolic modelling reveals cell-type specific epigenetic control points of the macrophage metabolic network. BMC genomics. 2015;16(1):809.
120. Schultz A, Qutub AA. Reconstruction of tissue-specific metabolic networks using CORDA. PLoS Comput Biol. 2016;12(3):e1004808.
121. Estévez SR, Nikoloski Z. Context-specific metabolic model extraction based on regularized least squares optimization. PloS one. 2015;10(7):e0131875.
122. Fyson N, Kim MK, Lun D, et al. Gene-centric constraint of metabolic models. bioRxiv. 2017;p. 116558.
123. Zhang SW, Gou WL, Li Y. Prediction of metabolic fluxes from gene expression data with Huber penalty convex optimization function. Molecular BioSystems. 2017;.
124. Guo W, Feng X. OM-FBA: Integrate Transcriptomics Data with Flux Balance Analysis to Decipher the Cell Metabolism. PloS one. 2016;11(4):e0154188.
125. Deutscher D, Meilijson I, Schuster S, et al. Can single knockouts accurately single out gene functions? BMC Systems Biology. 2008;2(1):50.

126. Nijman S. Synthetic lethality: general principles, utility and detection using genetic screens in human cells. FEBS letters. 2011;585(1):1–6.
127. Suthers PF, Zomorrodi A, Maranas CD. Genome-scale gene/reaction essentiality and synthetic lethality analysis. Molecular systems biology. 2009;5(1):301.
128. Pratapa A, Balachandran S, Raman K. Fast-SL: an efficient algorithm to identify synthetic lethal sets in metabolic networks. Bioinformatics. 2015;p. btv352.
129. Tobalina L, Pey J, Planes FJ. Direct calculation of minimal cut sets involving a specific reaction knock-out. Bioinformatics. 2016;32(13):2001–2007.
130. Jerby-Arnon L, Pfetzer N, Waldman YY, et al. Predicting cancer-specific vulnerability via data-driven detection of synthetic lethality. Cell. 2014;158(5):1199–1209.
131. Megchelenbrink W, Katzir R, Lu X, et al. Synthetic dosage lethality in the human metabolic network is highly predictive of tumor growth and cancer patient survival. Proceedings of the National Academy of Sciences. 2015;112(39):12217–12222.
132. McAnulty MJ, Yen JY, Freedman BG, et al. Genome-scale modeling using flux ratio constraints to enable metabolic engineering of clostridial metabolism in silico. BMC systems biology. 2012;6(1):42.
133. Yen JY, Nazem-Bokaee H, Freedman BG, et al. Deriving metabolic engineering strategies from genome-scale modeling with flux ratio constraints. Biotechnology journal. 2013;8(5):581–594.
134. Segre D, Vitkup D, Church GM. Analysis of optimality in natural and perturbed metabolic networks. Proceedings of the National Academy of Sciences. 2002;99(23):15112–15117.
135. Raval A, Ray A. Introduction to biological networks. CRC Press; 2013.
136. Yizhak K, Benyamini T, Liebermeister W, et al. Integrating quantitative proteomics and metabolomics with a genome-scale metabolic network model. Bioinformatics. 2010;26(12):i255–i260.
137. Shlomi T, Berkman O, Ruppin E. Regulatory on/off minimization of metabolic flux changes after genetic perturbations. Proceedings of the National Academy of Sciences of the United States of America. 2005;102(21):7695–7700.
138. Kim J, Reed JL. RELATCH: relative optimality in metabolic networks explains robust metabolic and regulatory responses to perturbations. Genome biology. 2012;13(9):1.
139. Angione C, Pratanwanich N, Lió P. A Hybrid of Metabolic Flux Analysis and Bayesian Factor Modeling for Multiomic Temporal Pathway Activation. ACS synthetic biology. 2015;4(8):880–889.
140. Oyetunde T, Czajka J, Wu G, et al. Metabolite patterns reveal regulatory responses to genetic perturbations. arXiv preprint arXiv:170101744. 2017;.
141. Knorr AL, Jain R, Srivastava R. Bayesian-based selection of metabolic objective functions. Bioinformatics. 2007;23(3):351–357.
142. Sendin J, Exler O, Banga JR. Multi-objective mixed integer strategy for the optimisation of biological networks. IET systems biology. 2010;4(3):236–248.
143. Xu G. An Iterative Strategy for Bi-objective Optimization of Metabolic Pathways. In: Proceedings of the 2011 Fourth International Joint Conference on Computational Sciences and Optimization. IEEE Computer Society; 2011. p. 587–588.
144. Angione C, Costanza J, Carapezza G, et al. Multi-target analysis and design of mitochondrial metabolism. PloS one. 2015;10(9):e0133825.
145. de Hijas-Liste GM, Klipp E, Balsa-Canto E, et al. Global dynamic optimization approach to predict activation in metabolic pathways. BMC systems biology. 2014;8(1):1.
146. Angione C, Carapezza G, Costanza J, et al. Pareto optimality in organelle energy metabolism analysis. IEEE/ACM transactions on computational biology and bioinformatics. 2013;10(4):1032–1044.
147. Xu M, Bhat S, Smith R, et al. Multi-objective optimisation of metabolic productivity and thermodynamic performance. Computers & Chemical Engineering. 2009;33(9):1438–1450.
148. Sendín OH, Vera J, Torres NV, et al. Model based optimization of biochemical systems using multiple objectives: a comparison of several solution strategies. Mathematical and Computer Modelling of Dynamical Systems. 2006;12(5):469–487.

149. Angione C, Costanza J, Carapezza G, et al. Analysis and design of molecular machines. Theoretical Computer Science. 2015;599:102–117.
150. Costanza J, Carapezza G, Angione C, et al. Robust design of microbial strains. Bioinformatics. 2012;28(23):3097–3104.
151. Deb K, Pratap A, Agarwal S, et al. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. Trans Evol Comp. 2002 Apr;6(2):182–197. Available from: http://dx.doi.org/10.1109/4235.996017.
152. Fortin FA, Parizeau M. Revisiting the NSGA-II crowding-distance computation. In: Proceedings of the 15th annual conference on Genetic and evolutionary computation. ACM; 2013. p. 623–630.
153. Zheng J, Shen R, Zou J. Enhancing diversity for NSGA-II in evolutionary multi-objective optimization. In: Natural Computation (ICNC), 2012 Eighth International Conference on. IEEE; 2012. p. 654–657.
154. Burgard AP, Pharkya P, Maranas CD. Optknock: a bilevel programming framework for identifying gene knockout strategies for microbial strain optimization. Biotechnology and bioengineering. 2003;84(6):647–657.
155. Pharkya P, Burgard AP, Maranas CD. Exploring the overproduction of amino acids using the bilevel optimization framework OptKnock. Biotechnology and bioengineering. 2003;84(7):887–899.
156. Tepper N, Shlomi T. Predicting metabolic engineering knockout strategies for chemical production: accounting for competing pathways. Bioinformatics. 2010;26(4):536–543.
157. Ranganathan S, Suthers PF, Maranas CD. OptForce: an optimization procedure for identifying all genetic manipulations leading to targeted overproductions. PLoS Comput Biol. 2010;6(4):e1000744.
158. Chowdhury A, Zomorrodi AR, Maranas CD. k-OptForce: integrating kinetics with flux balance analysis for strain design. PLoS Comput Biol. 2014;10(2):e1003487.
159. Xu Z, Zheng P, Sun J, et al. ReacKnock: identifying reaction deletion strategies for microbial strain optimization based on genome-scale metabolic network. PloS one. 2013;8(12):e72150.
160. Pharkya P, Burgard AP, Maranas CD. OptStrain: a computational framework for redesign of microbial production systems. Genome research. 2004;14(11):2367–2376.
161. Kim J, Reed JL, Maravelias CT. Large-scale bi-level strain design approaches and mixed-integer programming solution techniques. PLoS One. 2011;6(9):e24162.
162. Lun DS, Rockwell G, Guido NJ, et al. Large-scale identification of genetic design strategies using local search. molecular systems biology. 2009;5(1):296.
163. Fowler ZL, Gikandi WW, Koffas MA. Increased malonyl coenzyme A biosynthesis by tuning the Escherichia coli metabolic network and its application to flavanone production. Applied and environmental microbiology. 2009;75(18):5831–5839.
164. Rocha M, Maia P, Mendes R, et al. Natural computation meta-heuristics for the in silico optimization of microbial strains. BMC bioinformatics. 2008;9(1):1.
165. Nagrath D, Avila-Elchiver M, Berthiaume F, et al. Soft constraints-based multiobjective framework for flux balance analysis. Metabolic engineering. 2010;12(5):429–445.
166. Kelk SM, Olivier BG, Stougie L, et al. Optimal flux spaces of genome-scale stoichiometric models are determined by a few subnetworks. Scientific reports. 2012;2:580.
167. Maarleveld TR, Wortel MT, Olivier BG, et al. Interplay between constraints, objectives, and optimality for genome-scale stoichiometric models. PLoS Comput Biol. 2015;11(4):e1004166.
168. Zomorrodi AR, Maranas CD. OptCom: a multi-level optimization framework for the metabolic modeling and analysis of microbial communities. PLoS Comput Biol. 2012;8(2):e1002363.
169. Khandelwal RA, Olivier BG, Röling WF, et al. Community flux balance analysis for microbial consortia at balanced growth. PLoS One. 2013;8(5):e64567.
170. Gottstein W, Olivier BG, Bruggeman FJ, et al. Constraint-based stoichiometric modelling from single organisms to microbial communities. Journal of The Royal Society Interface. 2016;13(124):20160627.
171. Shoaie S, Ghaffari P, Kovatcheva-Datchary P, et al. Quantifying diet-induced metabolic changes of the human gut microbiome. Cell metabolism. 2015;22(2):320–331.

172. Louca S, Doebeli M. Calibration and analysis of genome-based models for microbial ecology. Elife. 2015;4:e08208.
173. Harcombe WR, Riehl WJ, Dukovski I, et al. Metabolic resource allocation in individual microbes determines ecosystem interactions and spatial dynamics. Cell reports. 2014;7(4):1104–1115.
174. Willemsen AM, Hendrickx DM, Hoefsloot HC, et al. MetDFBA: incorporating time-resolved metabolomics measurements into dynamic flux balance analysis. Molecular BioSystems. 2015;11(1):137–145.
175. Zomorrodi AR, Islam MM, Maranas CD. d-OptCom: dynamic multi-level and multi-objective metabolic modeling of microbial communities. ACS synthetic biology. 2014;3(4):247–257.
176. Balsa-Canto E, Henriques D, Gábor A, et al. AMIGO2, a toolbox for dynamic modeling, optimization and control in systems biology. Bioinformatics. 2016;p. btw411.
177. Tran LM, Rizk ML, Liao JC. Ensemble modeling of metabolic networks. Biophysical journal. 2008;95(12):5606–5617.
178. Lee Y, Rivera JGL, Liao JC. Ensemble Modeling for Robustness Analysis in engineering non-native metabolic pathways. Metabolic engineering. 2014;25:63–71.
179. Oh YG, Lee DY, Lee SY, et al. Multiobjective flux balancing using the NISE method for metabolic network analysis. Biotechnology progress. 2009;25(4):999–1008.
180. El Samad H, Khammash M, Homescu C, et al. Optimal performance of the heat-shock gene regulatory network. IFAC Proceedings Volumes. 2005;38(1):19–24.
181. Stracquadanio G, Umeton R, Papini A, et al. Analysis and optimization of c3 photosynthetic carbon metabolism. In: BioInformatics and BioEngineering (BIBE), 2010 IEEE International Conference on. IEEE; 2010. p. 44–51.
182. Zhang HX, Goutsias J. A comparison of approximation techniques for variance-based sensitivity analysis of biochemical reaction systems. BMC bioinformatics. 2010;11(1):1.
183. Petzold L, Li S, Cao Y, et al. Sensitivity analysis of differential-algebraic equations and partial differential equations. Computers & chemical engineering. 2006;30(10):1553–1559.
184. Kiparissides A, Hatzimanikatis V. Thermodynamics-based Metabolite Sensitivity Analysis in metabolic networks. Metabolic Engineering. 2016;.
185. Miskovic L, Hatzimanikatis V. Production of biofuels and biochemicals: in need of an ORACLE. Trends in biotechnology. 2010;28(8):391–397.
186. Andreozzi S, Miskovic L, Hatzimanikatis V. iSCHRUNK–In Silico Approach to Characterization and Reduction of Uncertainty in the Kinetic Models of Genome-scale Metabolic Networks. Metabolic engineering. 2016;33:158–168.
187. Teetor P. R Cookbook: Proven Recipes for Data Analysis, Statistics, and Graphics. O'Reilly Media; 2011.
188. Matloff N. The Art of R Programming: A Tour of Statistical Software Design. No Starch Press; 2011.
189. Deb K. Multi-objective optimization using evolutionary algorithms, 2001. Chicheter, John-Wiley.; 2001.
190. Sun S. A survey of multi-view machine learning. Neural Computing and Applications. 2013;23(7-8):2031–2038.
191. Frolova A, Obolenska M. Integrative approaches for data analysis in systems biology: Current advances. In: 2016 II International Young Scientists Forum on Applied Physics and Engineering (YSF); 2016. p. 194–198.
192. Ritchie MD, Holzinger ER, Li R, et al. Methods of integrating data to uncover genotype-phenotype interactions. Nature Reviews Genetics. 2015;16(2):85–97.
193. Serra A, Fratello M, Fortino V, et al. MVDA: a multi-view genomic data integration methodology. BMC bioinformatics. 2015;16(1):1.
194. Taskesen E, Huisman SM, Mahfouz A, et al. Pan-cancer subtyping in a 2D-map shows substructures that are driven by specific combinations of molecular characteristics. Scientific reports. 2016;6(24949):1–13.
195. Speicher NK, Pfeifer N. Integrating different data types by regularized unsupervised multiple kernel learning with application to cancer subtype discovery. Bioinformatics. 2015;31(12):i268–i275.

196. McLachlan GJ, Bean RW, Ng SK. Clustering. Bioinformatics: Structure, Function and Applications. 2008;p. 423–439.
197. Chen X, Xu X, Huang JZ, et al. TW-k-means: Automated two-level variable weighting clustering algorithm for multiview data. IEEE Transactions on Knowledge and Data Engineering. 2013 April;25(4):932–944.
198. Shen R, Olshen AB, Ladanyi M. Integrative clustering of multiple genomic data types using a joint latent variable model with application to breast and lung cancer subtype analysis. Bioinformatics. 2009;25(22):2906–2912.
199. Greene D, Cunningham P. A matrix factorization approach for integrating multiple data views. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer; 2009. p. 423–438.
200. Wang B, Mezlini AM, Demir F, et al. Similarity network fusion for aggregating data types on a genomic scale. Nature methods. 2014;11(3):333–337.
201. Angione C, Conway M, Lió P. Multiplex methods provide effective integration of multi-omic data in genome-scale models. BMC bioinformatics. 2016;17(4):257.
202. Wasito I, Istiqlal AN, Budi I. Data integration model for cancer subtype identification using Kernel Dimensionality Reduction-Support Vector Machine (KDR-SVM). In: 2012 7th International Conference on Computing and Convergence Technology (ICCCT); 2012. p. 876–880.
203. Li Y, Wu FX, Ngom A. A review on machine learning principles For multi-view biological data integration. Briefings in Bioinformatics. 2016;p. bbw113.
204. Breiman L. Random forests. Machine learning. 2001;45(1):5–32.
205. Fortino V, Kinaret P, Fyhrquist N, et al. A Robust and Accurate Method for Feature Selection and Prioritization from Multi-Class OMICs Data. PloS one. 2014;9(9):e107801.
206. Acharjee A, Ament Z, West JA, et al. Integration of metabolomics, lipidomics and clinical data using a machine learning method. BMC Bioinformatics. 2016;17(15):37.
207. Pratanwanich N, Lió P. Exploring the complexity of pathway–drug relationships using latent Dirichlet allocation. Computational biology and chemistry. 2014;53:144–152.

Full code and details of the R tutorial on genetic design by multi-objective optimisation

Here we present and describe the full code needed to conduct Genetic Design by Multi-Objective Optimisation in R. Note that here, in addition to the functionality described in the main text, we here cover the boilerplate code associated with loading libraries and preparing data, and the more length aspects of the NSGA-II procedure.

First, we need to load the appropriate libraries:

- `tidyverse` is a bundle of generic utilities;
- `stringr` is a string manipulation utility installed alongside `tidyverse`;
- `fbar` is a library for flux balance analysis;

```
library(tidyverse)
library(stringr)
library(fbar)
```

This code block downloads and reads in a model, then extracts the list of genes from the model. The model takes the form of a tabular list of reactions.

```
model <- read_tsv('https://git.io/v1YsM',
                  col_types = c('cccdddc'))

genes_in_model <- model$geneAssociation %>%
  str_split('[()|& ]+') %>%
  flatten_chr() %>%
  discard(is.na) %>%
  discard(~ str_length(.x)==0)
```

The evaluation function is where the actual metabolic simulations are performed. This has four main stages:

1. The gene-reaction associations (`geneAssociation`) are evaluated in the context of which genes are present in this iteration (`genome`), to give an `activation` value, which is an estimate of reaction rate.
2. The `activation` value is used to alter the upper and lower bounds on reaction rate (`uppbnd` and `lowbnd`), to push reaction rates towards the rate estimates.
3. We conduct a round of FBA, optimising for maximum biomass.
4. We fix the biomass production value to its maximum by altering the corresponding `uppbnd` and `lowbnd` to be near the `flux` (+/-1%).
5. With the biomass value fixed, we alter the objective coefficient (`obj_coef`) to target optimisation of the synthetic objective.

The technique of fixing the biomass followed by maximising the synthetic objective is important because there could still be slack in the model after the first optimisation stage, and we wish to have a reliable estimate of the synthetic objective.

```
evaluation_function <- function(genome){

  res <- model %>%
    mutate(activation = fbar::gene_eval(expressions = geneAssociation,
                                        genes = names(genome),
                                        presences = genome
                                        ),
           activation = coalesce(activation, 1),
           uppbnd = pmin(uppbnd, 1000*activation+0.1),
           lowbnd = pmax(lowbnd, -1000*activation-0.1)) %>%
    fbar::find_fluxes_df(do_minimization = FALSE) %>%
    mutate(lowbnd = ifelse(abbreviation=='Biomass_Ecoli_core_w/GAM',
                           flux*0.99,
                           lowbnd),
           uppbnd = ifelse(abbreviation=='Biomass_Ecoli_core_w/GAM',
                           flux*1.01,
                           uppbnd),
           obj_coef = 1*(abbreviation=='EX_ac(e)')) %>%
    fbar::find_fluxes_df(do_minimization = FALSE)
```

```
  return(list(bm = filter(res, abbreviation=='Biomass_Ecoli_core_w/GAM')$flux,
              synth = filter(res, abbreviation=='EX_ac(e)')$flux))
}
```

Non-domination sorting is the first stage of the selection procedure in NSGA-II. The code might seem quite opaque, but the idea is as follows:

1. We perform an `inner_join` in order to compare every point against every other point.
2. For each point (`id.x`), we see if there exists any second point (`id.y`) that has a higher value in all objectives. Where such a second point exists, we term the original point 'dominated'.
3. We find the set of points that have no dominating point, and term this the first non-dominated front.
4. We repeat this procedure, but ignore points in the first non-dominated front to find the second non-dominated front, and so on.

```
non_dom_sort <- function(input){
  input_long <- input %>%
    gather(property, value, -id) %>%
    mutate(front=NA)

  currentfront <- 1

  while(any(is.na(input_long$front))){

    input_long <- input_long %>%
      inner_join(.,., by='property') %>%
      group_by(id.x,id.y) %>%
      mutate(dominance = ifelse(all(value.x>=value.y),
                                'xdomy',
                                ifelse(all(value.y>=value.x),
                                       'ydomx',
                                       'nondom'
                                       )
                                )
      ) %>%
      group_by(id.x) %>%
      mutate(front = ifelse(all(dominance[is.na(front.y)] %in% c('xdomy', 'nondom')),
                            pmin(currentfront, front.x, na.rm=TRUE),
                            NA
                            )
      ) %>%
      group_by(id = id.x, property = property, front, value = value.x) %>%
      summarise

    currentfront <- currentfront + 1
  }

  return(
```

```
    input_long %>%
      spread(property, value)
  )

}
```

The second part of the NSGA-II evaluation procedure is finding the crowding distance. This is used to break ties between points in the same non-dominated front. For each front and for each dimension, this function sorts the points into order along the dimension, and finds the normalised distance between the proceeding point and succeeding point. These values are summed up across each dimension to find the value for the point.

```
crowding_distance <- function(input){
  return(
    input %>%
      gather(property, value, -id, -front) %>%
      group_by(front, property) %>%
      arrange(value) %>%
      mutate(crowding = (lead(value)-lag(value))/(max(value)-min(value)),
             crowding = ifelse(is.na(crowding),Inf, crowding)) %>%
      group_by(id) %>%
      mutate(crowding = sum(crowding)) %>%
      spread(property, value)
  )
}
```

The following code is the genetic loop of the algorithm. It is explained by code comments, but follows a normal pattern of evaluating, sorting, selecting from and mutating the population. The genetic algorithm used here is a modified version of NSGA-II [151], with a population of 200 individuals and carrying out 500 iterations.

```
start_genome <- set_names(rep_along(genes_in_model, TRUE), genes_in_model)
pop <- list(start_genome)

popsize = 200
generations = 500

pb <- txtProgressBar(max=generations, style=3)
for(i in 1:generations){
  setTxtProgressBar(pb, i)
  results <- map_df(pop, evaluation_function) %>% # Evaluate all the genomes
    mutate(bm=signif(bm), synth=signif(synth)) %>% # Round results
    mutate(id = 1:n()) %>% # label the results
    sample_frac() %>% # Shuffle
    non_dom_sort() %>% # Find the non-dominated fronts
    crowding_distance() %>% # Find the crowding distances
    arrange(front, desc(crowding)) # Sort by front, breaking ties by crowding distance

  selected <- results %>%
    filter(row_number() <= popsize/2) %>% # Keep the best half of the population
```

```
      getElement('id')

    kept_pop <- pop[selected]

    altered_pop <- kept_pop %>%
      sample(popsize-length(selected), TRUE) %>% # Sample parents from population
      map(function(genome){
        xor(genome, runif(length(genome))>0.98) # Mutate parents to create offspring
        })

    pop <- unique(c(kept_pop, altered_pop)) # Combine the offspring and parent populations
  }
```

Once we have a results set, we can construct a plot to view the non-dominated fronts. We can see how the first front describes the trade-off between biomass and the synthetic objective, with the lines showing the dominated area (to the bottom left).

```
pop %>%
  arrange(desc(front)) %>%
  ggplot(aes(x=bm, y=synth, colour=factor(front))) +
  geom_point() + geom_step(direction='vh', alpha=0.5) +
  theme_bw()
```