BMC
Systems Biology

METHODOLOGY ARTICLE

Open Access

Designing optimal cell factories: integer programming couples elementary mode analysis with regulation

Christian Jungreuthmayer[1,2] and Jürgen Zanghellini[1,2]*

Abstract

Background: Elementary mode (EM) analysis is ideally suited for metabolic engineering as it allows for an unbiased decomposition of metabolic networks in biologically meaningful pathways. Recently, constrained minimal cut sets (cMCS) have been introduced to derive optimal design strategies for strain improvement by using the full potential of EM analysis. However, this approach does not allow for the inclusion of regulatory information.

Results: Here we present an alternative, novel and simple method for the prediction of cMCS, which allows to account for boolean transcriptional regulation. We use binary linear programming and show that the design of a regulated, optimal metabolic network of minimal functionality can be formulated as a standard optimization problem, where EM and regulation show up as constraints. We validated our tool by optimizing ethanol production in *E. coli*. Our study showed that up to 70% of the predicted cMCS contained non-enzymatic, non-annotated reactions, which are difficult to engineer. These cMCS are automatically excluded by our approach utilizing simple weight functions. Finally, due to efficient preprocessing, the binary program remains computationally feasible.

Conclusions: We used integer programming to predict efficient deletion strategies to metabolically engineer a production organism. Our formulation utilizes the full potential of cMCS but adds additional flexibility to the design process. In particular our method allows to integrate regulatory information into the metabolic design process and explicitly favors experimentally feasible deletions. Our method remains manageable even if millions or potentially billions of EM enter the analysis. We demonstrated that our approach is able to correctly predict the most efficient designs for ethanol production in *E. coli*.

Keywords: Metabolic engineering, Elementary modes, Minimal cut sets, Integer programming, Strain optimization, Ethanol production, Minimal functionality, Gene regulation

Background

Arguably the most successful methods in computer aided strain design are based on constraint-based modeling [1]. These methods allow to predict phenotypes by calculating steady state flux distributions through a metabolic network (typically using some kind of flux balance analysis [2]). Various algorithms allow searching for combinatorial gene deletion strategies to optimize the production efficiency of strains [3-6]. These methods utilize an optimization principle, which has been shown to give accurate

predictions in wild type strains. Typically, evolutionary rationalized objectives like maximization of biomass or minimization of metabolic adjustments are used to predict changes in the flux distribution. However, these objectives become more problematic with an increasing number of gene deletions as the engineered strains have no time to adapt and thus are far from an evolutionary optimum [7].

An alternative way of predicting optimal strain design is to use elementary mode analysis (EMA) [8-13]. EMA allows decomposing a complex metabolic network into unique and biologically meaningful pathways, called elementary modes (EM) [14,15]. An EM is a minimal, and indivisible set of reactions that operates under steady state

*Correspondence: juergen.zanghellini@acib.at
[1] Austrian Centre of Industrial Biotechnology, Vienna, Austria
[2] Department of Biotechnology, University of Natural Resources and Life Sciences, Vienna, Austria

conditions, while obeying all (ir-)reversibility constraints on the reactions. EM are minimal in the sense that knocking out any one of their contributing reaction will exclude the whole mode from carrying any steady state flux. The entire set of EM, however, describes the full metabolic potential of a cell in an unbiased way. By iteratively deleting EM with unwanted properties a metabolic network of minimal functionality (NMF) can be generated [12]. This procedure, however, does not necessarily return the NMF with the minimum number of deletions.

A rigorous formulation – constrained minimal cut sets (cMCS) – for generating NMF has recently been put forward [16]. It relies on the concept of minimal cut sets (MCS). These are (minimal) sets of deletions, which block undesirable network functionality, like the secretion of unwanted by-products. cMCS allow to keep desirable network properties while simultaneously disabling unwanted functionality [16]. Thus cMCS are ideally suited to design NMF. Moreover, with cMCS it is possible not only to derive the minimal necessary number of metabolic interventions but also to exhaustively predict all possible combinations of deletions resulting in identical NMF.

Here we present an alternative formulation to predict the optimal engineering strategy for the design of MNF. We formulate an optimization problem and show that cMCS can be easily calculated by binary linear programming (BLP) for which commercial and non-commercial solvers are readily available. The scope of our approach is similar to the algorithm presented by [16] but it is more flexible and – most importantly – it allows to include regulatory information in the design process of rational engineering strategies. Static gene regulatory rules can be considered as long as they are formulated in boolean logic terms.

Theory
Definitions
We consider the standard steady-state problem of a metabolic network with m internal metabolites and n reactions, i.e. $S \cdot \hat{v} = 0$. Here, S denotes the $m \times n$ stoichiometric matrix of the network, and $\hat{v}$ the n-dimensional flux vector through the network.

Let $\hat{e}$ be an EM flux vector [14,15] fulfilling the steady state condition, and $e = e(\hat{e})$ its binary representation,

$$e_i := e(\hat{e}_i) = \begin{cases} 1 & \text{if } \hat{e}_i \neq 0 \\ 0 & \text{if } \hat{e}_i = 0 \end{cases}, \quad i = \{1, \ldots, n\}. \tag{1}$$

e_i indicates whether reaction i is part of the EM $\hat{e}$. That is, $e_i = 1$ if and only if a reaction is carrying flux either in forward or backward direction. Similar to equation (1) let v denote the binary representation of any valid flux distribution $\hat{v}$. Then the product

$$e^{\mathrm{T}} v \leq e^{\mathrm{T}} e = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} e_i =: ||e||, \tag{2}$$

indicates if e is part of v as the equality only holds when all "active" reactions in e are also carrying flux in v.

Finally, we group all q binary EM of S into three matrices

$$G := (e_1, \ldots, e_r)^{\mathrm{T}}, \tag{3a}$$

$$H := (e_{r+1}, \ldots, e_{r+s})^{\mathrm{T}}, \tag{3b}$$

$$K := (e_{r+s+1}, \ldots, e_{r+s+t})^{\mathrm{T}}. \tag{3c}$$

where $q = r + s + t$, as all EM are in one of the three matrices. The "goal matrix", G, contains all desirable EM, which define the minimal properties of the NMF and must therefore be kept. The "kill matrix", K, consists of the unwanted EM, which must not be part of the final flux space and have to be deleted from the network. Finally, the helper matrix, H holds all remaining EM. These modes do not affect the primary design criterion, and therefore may or may not be present in the final design.

In the notation of Hädicke and Klamt [16], our kill matrix K is their set of target modes $\mathbf{T}$. Our G is a subset of their set of desired modes $\mathbf{D}$. We collect all other modes in H, while they split these EM between the sets of desired modes, $\mathbf{D}$, and the sets of neutral modes. In their formulation Hädicke and Klamt [16] aim to keep at least n desired EM out of all modes in $\mathbf{D}$. These "surviving" EM build our G. If, however, $|\mathbf{D}| = n$ then $\mathbf{D} = G$ and hence, both definitions are identical.

Minimum number of deletions, Δ_{min}
By setting up a BLP problem, equation (2) may be used to predict the minimal set of knockouts to stop any given set of EM, i.e. the K-matrix, contributing to the steady state flux distribution

$$\max \quad ||x|| \tag{4a}$$

$$\text{s.t.} \quad Gx = |g| \tag{4b}$$

$$Hx \leq |h| \tag{4c}$$

$$Kx \leq |k| - \mathbf{1} \tag{4d}$$

$$x = (x_1, \ldots, x_n)^{\mathrm{T}}, \quad x_i \in \{0, 1\} \forall i, \tag{4e}$$

We used $|g| = (||e_1||, \ldots, ||e_r||)^{\mathrm{T}}$, $|h| = (||e_{r+1}||, \ldots, ||e_{r+s}||)^{\mathrm{T}}$, and $|k| = (||e_{r+s+1}||, \ldots, ||e_{r+s+t}||)^{\mathrm{T}}$ to denote the vector of norms of each row of the matrix G, H, and K, respectively. $\mathbf{1} = (1, \ldots, 1)$ represents a vector of ones. The solution vector x, is the binary representation of all reactions participating in the designed NMF. Equation (4) is indeed a BLP problem as x is binary and $||x|| = \sum_{i=1}^{n} x_i$ is linear.

In equation (4) we used a matrix formulation, which is shorthand for the optimization problem in terms of all $q = r + s + t$ binary EM vectors $\boldsymbol{e}_i$,

$$\max \quad ||\boldsymbol{x}||$$

$$\text{s.t.} \quad \boldsymbol{e}_g^{\mathrm{T}} \boldsymbol{x} = ||\boldsymbol{e}_g||, \quad g \in \{1, \ldots, r\}$$

$$\boldsymbol{e}_h^{\mathrm{T}} \boldsymbol{x} \leq ||\boldsymbol{e}_h||, \quad h \in \{r+1, \ldots, r+s\}$$

$$\boldsymbol{e}_k^{\mathrm{T}} \boldsymbol{x} \leq ||\boldsymbol{e}_k|| - 1, \quad k \in \{r+s+1, \ldots, r+s+t\}.$$

Here we used indices g, h, k as a reminder that these EM vectors are the rows of the matrices $\boldsymbol{G}, \boldsymbol{H}$, and $\boldsymbol{K}$, respectively. Note that each EM acts as a constraint for the optimization problem.

To understand equation (4) we recall that the binary solution vector $\boldsymbol{x}$ indicates whether a reaction i can carry flux ($x_i = 1$) or needs to be deleted ($x_i = 0$). Thus maximizing the norm of $\boldsymbol{x}$, corresponds to minimizing the number of knockouts. The constraint (4b) requires that any solution includes all desired EM as – according to equation (2) – only then the product $\boldsymbol{e}_i^{\mathrm{T}} \boldsymbol{x}$ is limited by the norm of $\boldsymbol{e}_i$. Similar, equation (4d) demands that its solutions are at least one active reaction short, i.e. has more zeros than any EM in $\boldsymbol{K}$. As already one single knockout in an EM kills it, these modes will not contribute to the desired design. Finally, constraint (4c) states that the EM of $\boldsymbol{H}$ may be included in the solution. In fact, the inequality (4c) does not constrain the system in any way. Equation (4c) is merely included for the sake of accounting completely for all EM in the network.

The minimal number of deletions can then be determined easily by counting the number of zeros in the calculated solution $\boldsymbol{x}$,

$$\Delta_{\min} = n - ||\boldsymbol{x}||. \tag{5}$$

Predicting all optimal sets of deletions

Equation (4) may either have no or a finite number of solutions. In the first case, no knockout strategy accommodates all constraints. However, if the constraints are relaxed, i.e. EM are shifted from $\boldsymbol{G}$ to either $\boldsymbol{H}$ or $\boldsymbol{K}$ [the limit being $\boldsymbol{G} = (\boldsymbol{e}_1)^{\mathrm{T}}, \boldsymbol{H} = \boldsymbol{0}$, and $\boldsymbol{K} = (\boldsymbol{e}_2, \ldots, \boldsymbol{e}_q)^{\mathrm{T}}$], it is always possible to find at least one solution.

Alternate optimal solutions may be found by successively excluding already existing solutions $\boldsymbol{x}^{(j)}$ of equation (4) by adding [17,18],

$$\sum_{i \in B} x_i - \sum_{i \in N} x_i \leq |B| - 1, \tag{6a}$$

$$B = \{i | x_i^{(j)} = 1\}, \quad N = \{i | x_i^{(j)} = 0\}. \tag{6b}$$

Note that repeatedly applying equation (6) will not only generate all sets of different minimal knockouts but also enumerate all other solutions sorted by the number of deletions.

The final sequence contains all possible solutions. It also contains "inefficient" or non-minimal solutions. Consider a series of two reactions, $A \rightarrow B, B \rightarrow C$. To suppress the production of C, the knocking out of either reaction suffices. Knocking out both is admissible, although inefficient. To avoid calculating non-minimal solutions we split equation (6) into two constraints,

$$\sum_{i \in B} x_i \leq |B| - 1, \tag{7a}$$

$$\sum_{i \in N} x_i \geq 1. \tag{7b}$$

In matrix notation these constraints read

$$[\boldsymbol{x}^{(j)}]^{\mathrm{T}} \boldsymbol{x} \leq ||\boldsymbol{x}^{(j)}|| - 1 \tag{8a}$$

$$[\boldsymbol{1} - \boldsymbol{x}^{(j)}]^{\mathrm{T}} \boldsymbol{x} \geq 1. \tag{8b}$$

The first excludes already existing solutions, $\boldsymbol{x}^{(j)}$, the second ensures that all solutions will be minimal. In other words, no supersets of already determined solutions will be calculated.

It is possible to influence the succession of solutions by adding weights w_i to the objective function. Rather than maximizing $||\boldsymbol{x}||$ in equation (4a) we may use

$$\max \; ||\boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}||, \tag{9}$$

with $\boldsymbol{w}^{\mathrm{T}} = (w_1, \ldots, w_n)$. This allows to easily distinguish chemical from genetic interventions. If uptake reactions are assigned a small and all other reactions a large weight, our algorithm will favor deletions in the uptake reactions as they contribute little to the objective function. Deleting uptake reactions can simply be achieved by removing the substrate from the culture medium. We give guidelines for the choice of reaction weights in the example below.

Illustrative example

To illustrate our algorithm we will use the toy network shown in Figure 1. The complete set of EM and their binary representation are listed in Table 1, and illustrated in Figure 2.

Suppose we use A as feed stock and want to engineer the conversion of A into P. Our aim is to maximize the utilization of A for the efficient production of P. According to Table 1, $\hat{\boldsymbol{e}}_1^{\mathrm{T}}$ is the only mode which maximizes utilization of A, while efficiently producing P. Hence the goal matrix $\boldsymbol{G}$ is simply given by $\boldsymbol{e}_1$. $\hat{\boldsymbol{e}}_2^{\mathrm{T}}$, and $\hat{\boldsymbol{e}}_3^{\mathrm{T}}$ inefficiently synthesize P. $\hat{\boldsymbol{e}}_4^{\mathrm{T}}, \hat{\boldsymbol{e}}_5^{\mathrm{T}}$, and $\hat{\boldsymbol{e}}_6^{\mathrm{T}}$ sub-optimally utilize A. These modes need to be deleted and therefore populate $\boldsymbol{K}$. The remaining EM do not utilize A. It is irrelevant whether or not those modes are present in the final design as they will have no negative impact. Thus the full BLP problem
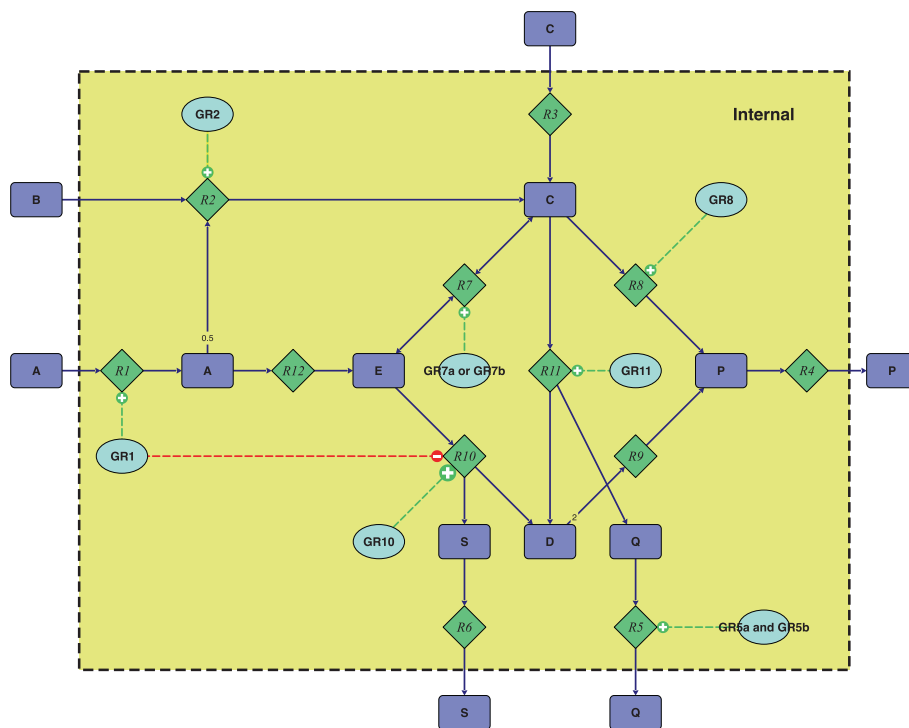
Figure 1 Illustrative example network. Illustrative example network containing the metabolites A to E, P, Q and S, the reactions R1 to R12, and the genes GR1, GR2, GR5a, GR5b, GR7a, GR7b, GR8, GR10, and GR11. All reactions are irreversible, except for R7. Transition from E to C is defined as the forward direction of R7. Small numbers in the edges of reactions indicate stoichiometric coefficients, if they are different from one. All metabolites inside the shaded area are considered internal and are subject to the steady state condition. Gene-enzyme-reaction mapping is indicated by dashed lines. Reaction R5 is catalyzed by an enzyme complex encoded by gene GR5a and GR5b. Reaction R7 is catalyzed by two enzymes encoded by GR7a or GR7b. The reaction R10 is catalyzed by GR10. However, activity of R10 is inhibited if GR1 is expressed. For the reaction R3, R4, R6, R9 and R12 no gene-enzyme-reaction mapping is available.

is defined by the matrices and vectors listed in Table 1. Explicitly, equation (4) reads

$$\max \sum_{i=1}^{12} x_i$$

subject to

$$x_1 + x_4 + x_7 + x_8 + x_{12} = 5$$
$$x_3 + x_4 + x_8 \leq 3$$
$$x_3 + x_4 + x_5 + x_9 + x_{11} \leq 5$$
$$x_3 + x_4 + x_6 + x_7 + x_9 + x_{10} \leq 6$$
$$x_1 + x_4 + x_5 + x_7 + x_8 + x_9 + x_{11} + x_{12} \leq 7$$
$$x_1 + x_4 + x_6 + x_9 + x_{10} + x_{12} \leq 5$$
$$x_1 + x_2 + x_4 + x_8 \leq 3$$
$$x_1 + x_2 + x_4 + x_5 + x_9 + x_{11} \leq 5$$
$$x_1 + x_2 + x_4 + x_6 + x_7 + x_9 + x_{10} \leq 6$$

The BLP returns the solution (given in vector notation),

$$x^{(1)} = (\, 1 \; 0 \; 1 \; 1 \; 1 \; 1 \; 1 \; 1 \; 0 \; 1 \; 1 \; 1 \,)^{\mathrm{T}}.$$

R2-R9 is the smallest possible MCS to achieve the design criterion. With the solution $x^{(1)}$ at hand we use equation (6b) to get the set of indices for the undeleted and deleted reactions, $B = \{1, 3, 4, 5, 6, 7, 8, 10, 11, 12\}$, and, $N = \{2, 9\}$, respectively. By adding the constraint equation (7),

$$x_1 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + x_{10} + x_{11} + x_{12} \leq 9,$$

$$x_2 + x_9 \geq 1,$$

to the equations above and resolving the problem, an alternative MCS may be calculated. An overview of all MCS is given in Table 2.

The calculated solutions do not take gene-enzyme-reaction mapping into account. As indicated in Figure 1, implementing the smallest MCS (R2 and R9) is infeasible, due to missing genetic information for R9. To account for biological feasibility we reevaluate the BLP problem using the weight function,

$$w_2^{\mathrm{T}} = (\, 0.1 \; 0.1 \; 0.1 \; 99 \; 1 \; 99 \; 2 \; 1 \; 99 \; 1 \; 1 \; 99 \,).$$

Here we assigned small weights (0.1) to uptake reactions (R1 to R3), which are easy to "delete" by removing the

Table 1 list of all EM for Figure 1

	EM flux vector, $\hat{e}_i$													Binary representation, e_i, of EM flux vector, $\hat{e}_i$														
	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12		R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	$\|\|e_i\|\|$	
EM 1	1.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	1.0	$G=$	1	0	0	1	0	0	1	1	0	0	0	1	5	$=\|g\|$
EM 2	1.0	0.0	0.0	0.5	1.0	0.0	1.0	0.0	0.5	0.0	1.0	1.0		1	0	0	1	1	0	1	1	1	0	1	1	8	
EM 3	1.0	0.0	0.0	0.5	0.0	1.0	0.0	0.0	0.5	1.0	0.0	1.0		1	0	0	1	0	1	0	0	1	1	0	1	6	
EM 4	0.5	1.0	0.0	1.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	$K=$	1	1	0	1	0	0	0	1	0	0	0	0	4	$=\|k\|$
EM 5	0.5	1.0	0.0	0.5	1.0	0.0	0.0	0.0	0.5	0.0	1.0	0.0		1	1	0	1	1	0	0	0	1	0	1	0	6	
EM 6	0.5	1.0	0.0	0.5	0.0	1.0	-1.0	0.0	0.5	1.0	0.0	0.0		1	1	0	1	0	1	1	0	1	1	0	0	7	
EM 7	0.0	0.0	1.0	1.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0		0	0	1	1	0	0	0	1	0	0	0	0	3	
EM 8	0.0	0.0	1.0	0.5	1.0	0.0	0.0	0.0	0.5	0.0	1.0	0.0	$H=$	0	0	1	1	1	0	0	0	1	0	1	0	5	$=\|h\|$
EM 9	0.0	0.0	1.0	0.5	0.0	1.0	-1.0	0.0	0.5	1.0	0.0	0.0		0	0	1	1	0	1	1	0	1	1	0	0	6	

List of all EM flux vectors, $\hat{e}_i$, and their binary representation, e_i, for the toy network illustrated in Figure 1. EM are sorted by decreasing order of substrate utilization of A. The matrices and vectors G, H, K, and $|g|, |h|, |k|$, respectively, are defined as used in the illustrative example of section "Illustrative example".
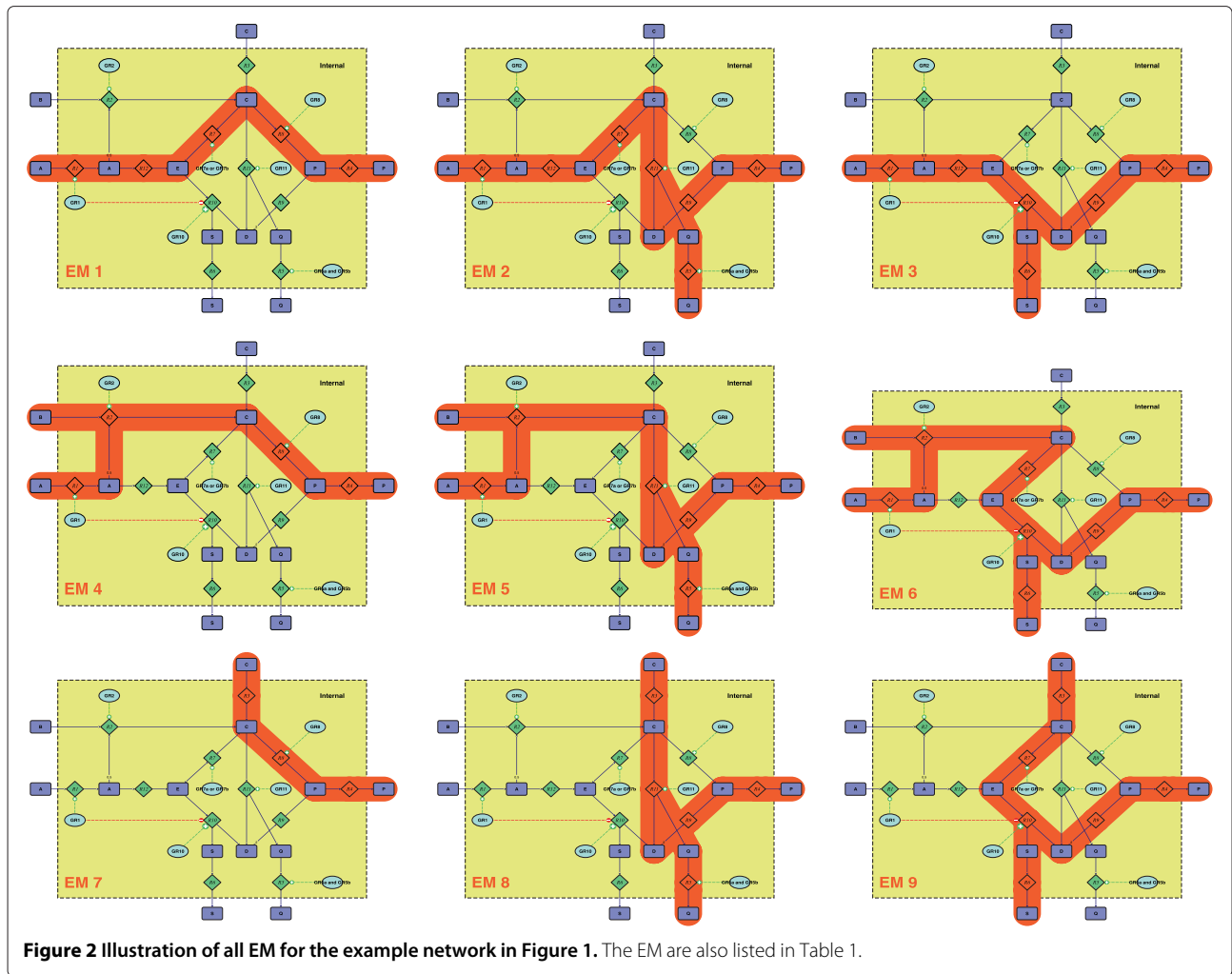
Figure 2 Illustration of all EM for the example network in Figure 1. The EM are also listed in Table 1.

corresponding substrate from the growth medium. Reactions with missing genetic information (R4, R6, R9, R12) received high weights (99), which made them "harder" to delete. Note that R3 is also lacking genetic information. Since it is an easily "deletable" uptake reaction, R3 was weighted with 0.1 rather than 99. We associated R7 with a weight of two as this reaction is catalyzed by two independent enzymes. On the other hand, R5 retained its weight of one as the reaction is catalyzed by a single enzyme complex encoded by two genes. The sequence of all possible MCS is listed in Table 2. Note that by using weight functions, experimentally implementable engineering strategies are predicted first. All other solutions are predicted, too. However, the weight function is able to account for experimental difficulties in implementing a reaction deletion *in vivo*.

In general, we assign reaction weights according to the number of independent enzymes or enzyme complexes catalyzing a reaction in parallel. Uptake reactions, however, should be favored over genetic deletions. Therefore

the sum of all weights for uptake reactions should be smaller than the smallest weight of the non-uptake reactions. On the other hand the weight for a "non-deletable" reaction (i.e. a reaction without genetic information)

Table 2 List of all MCS for Figure 1

	w_1					w_2				
i	minimal cut set			f_i		minimal cut set			f_i	
1	R2	R9		9.0		R2	R5	R10	301.2	*
2	R2	R5	R6	8.0		R2	R10	R11	301.2	*
3	R2	R5	R10	8.0	*	R2	R9		204.2	
4	R2	R6	R11	8.0		R2	R5	R6	203.2	
5	R2	R10	R11	8.0	*	R2	R6	R11	203.2	

List of all MCS for the most efficient production of P from A in the network Figure 1. Two different weight vectors were used, $w^T_1 = (1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1)$, and $w^T_2 = (0.1\ 0.1\ 0.1\ 99\ 1\ 99\ 2\ 1\ 99\ 1\ 1\ 99)$. MCS are sorted in decreasing order of the objective function $f_i = w^T_j\ x^{(i)}, j = \{1,2\}$ as calculated by our algorithm. (The sequence of MCS with equal objective value may differ depending on the BLP algorithm.) * marks MCS for which full genetic information is available.

should be larger than the sum of all other "deletable" reactions.

Including regulation

In the following we demonstrate the inclusion of boolean regulation by way of example. Typically, regulatory information is represented in logic statements [19] which may readily be added to equation (4). In Figure 1 we illustrate typical gene-enzyme-reaction mappings, like reactions catalyzed by single enzymes (G↦R), by multiple enzymes in parallel [(Ga OR Gb)↦R], or by single enzyme complexes [(Ga AND Gb)↦R]. As demonstrated, these interactions may be incorporated in weight functions. By adding appropriate constraints, BLP also allows the integration of inhibitions, like (NOT G)↦R; [Ga AND (NOT Gb)]↦R. For example a single gene-enzyme-reaction mapping, G↦R, is easily converted into the BLP constraint, $G - R = 0$. Similarly, the negation (NOT G)↦R transforms into $G + R = 1$. In Table 3 we summarize other interactions along with their constraint based formulation. An extension to more interaction partners is straight forward. More specifically, we list the regulatory constraints for the network in Figure 1 in Table 4.

Adding the regulatory constraints in Table 4 we maximize the BLP problem equation (4) using $||w_2^T x|| + ||y||$ as objective. Here

$$y = \left(y_1 \ y_2 \ y_{5a} \ y_{5b} \ y_{7a} \ y_{7b} \ y_8 \ y_{10} \ y_{11} \right)^T$$

denotes the binary vector of the involved genes. Note that integrating regulation into our algorithm only requires additional constraints and an extended objective function. This is in contrast to the original cMCS-method [16]. cMCS requires an independent, separate preprocessing step first to identify and remove all EM, which are in contradiction to regulatory constraints. Only then, cMCS can be applied. BLP, however, allows simultaneously integrating stoichiometric and regulatory constraints in a unified

framework. Moreover, BLP allows to fully consider reconstructed transcriptional regulatory networks.

Note that by using $||w_2^T x|| + ||y||$ as objective, we optimize for the combined effect of both, reactions and genes. Thus our objective predicts interventions with the smallest overall impact first. Again, it is possible to influence the succession of solutions by using weight functions for genes as well. However, this has not been investigated.

In Table 5 we collect all MCS to the regulatory BLP problem for the network in Figure 1. Note that the MCS 1 and 2 do not differ in terms of reactions but in terms of the deleted genes. All feasible MCS require two deletions at the genetic level, but three reaction deletions. The third reaction (R10) is suppressed due to GR1, rather than deleted. According to the design criterion GR1 is expressed in all desired EM. Thus all solutions to the BLP problem will necessarily be characterized by a down regulated R10. This reduces the total number of different MCS (in terms of reactions) from five to three (compare Table 2 and Table 5). Note that the MCS R2-R5-R6 and R2-R6-R11 of Table 2 are not MCS for the regulated system. As in the regulated system R10 is always suppressed, deletion of R6 becomes redundant. For the regulated network R2-R5-R6 and R2-R6-R11 are only cut sets, rather than MCS.

Optimizing metabolic functionality

All solutions to equation (4) and (6) are characterized by the smallest possible number of knockouts. However, their metabolic functionality may differ. This can be the case if $H \neq 0$, as individual EM from the helper matrix may be added or removed. With all optimal solutions at hand it is easy to pick those which additionally optimize the number of "surviving" EM. That is, we may look for solutions with the smallest/largest set of EM contributing to the metabolic functionality. However, for these questions it is not necessary to fully enumerate all solutions of equation (4). The answer is accessible by BLP as well.

Table 3 Truth table for the conversion of regulatory functions into constraints for BLP

G_a	G_b	R	(Ga OR Gb)↦R / $-1 \leq G_a + G_b - 2R \leq 0$	[(NOT Ga) OR Gb]↦R / $0 \leq G_a + G_b - 2R \leq 1$	(Ga AND Gb)↦R / $0 \leq G_a + G_b - 2R \leq 1$	[(NOT Ga) AND Gb]↦R / $-1 \leq -G_a + G_b - 2R \leq 0$
0	0	0	0	*	0	0
0	0	1	*	-2	*	*
0	1	0	*	*	1	*
0	1	1	-1	-1	*	-1
1	0	0	*	-1	1	-1
1	0	1	0	*	*	*
1	1	0	*	*	*	0
1	1	1	0	-2	0	*

*Marks values outside the constraint range.

Table 4 Regulatory constraints in Figure 1 for use in BLP

$y_1 - x_1 = 0$	$y_{5a} + y_{5b} - 2x_5 \leq 1$
$-y_1 + y_{10} - 2x_{10} \geq -1$	$y_{7a} + y_{7b} - 2x_7 \geq -1$
$-y_1 + y_{10} - 2x_{10} \leq 0$	$y_{7a} + y_{7b} - 2x_7 \leq 0$
$y_2 - x_2 = 0$	$y_8 - x_8 = 0$
$y_{5a} + y_{5b} - 2x_5 \geq 0$	$y_{11} - x_{11} = 0$

Regulatory constraints for use in BLP for the metabolic network in Figure 1. Here x_i, and y_j denote reactions and genes, respectively.

Let

$$p(\boldsymbol{e}) = \prod_{i \in C} e_i, \quad C = \{i | e_i = 1\}, \tag{10}$$

be the product of all reactions contributing to an EM. $p(\boldsymbol{e}) = 0$ if any reaction contributing to the EM e_i is knocked out, and 1 otherwise. Thus $p(\boldsymbol{e})$ indicates whether an EM contributes to the final steady state. Optimizing the number of surviving EM means we maximize (minimize) the number of participating EM,

$$\min/\max \sum_i p(\boldsymbol{e}_i), \quad i = \{r+1, \ldots, r+s\}. \tag{11}$$

Here i runs over all EM which may contribute to the steady state, i.e. over all modes stored in $\boldsymbol{H}$. Although $p_i = p(\boldsymbol{e}_i)$ is a product of binary variables, it is convertible into BLP using standard transformation rules [20] yielding

$$\min/\max \quad ||\boldsymbol{p}|| \tag{12a}$$
$$\text{s.t.} \quad \boldsymbol{Hx} \geq \boldsymbol{p}.*|\boldsymbol{h}| \tag{12b}$$
$$\boldsymbol{Hx} \leq \boldsymbol{p} + |\boldsymbol{h}| - 1 \tag{12c}$$
$$\boldsymbol{p} = (p_{r+1}, \ldots, p_{r+s})^{\mathrm{T}}, \quad \in \{0,1\} \forall i \tag{12d}$$
$$\text{equation (4b) to equation (4e)}, \tag{12e}$$

where we used the MATLAB notation for array multiplication ".*" to denote the element-wise product of the vectors $\boldsymbol{p}$ and $|\boldsymbol{h}|$.

Table 5 List of all MCS for the regulatory BLP in Figure 1

i	Gene deletion		Reaction deletion			f_i	
1	GR2	GR5a	R2	R5	R10	308.2	*
2	GR2	GR5b	R2	R5	R10	308.2	*
3	GR2	GR11	R2	R10	R11	308.2	*
4	GR2		R2	R9	R10	211.2	

List of all MCS for the regulatory BLP. MCS are sorted in decreasing order of the objective function $f_i = w^{\mathrm{T}}_2 x^{(i)} + ||y^{(i)}||$ as calculated by our algorithm. (The sequence of MCS with equal objective value may differ depending on the BLP algorithm.) * marks MCS for which full genetic information is available. MCS are split in the gene deletion part and the reaction deletion part. Note that the first three MCS require deletions of two genes. The corresponding reaction deletions are a consequence of those deletions. MCS 4 however, is not fully annotated (noticeable in the drop of f_i), and would require the deletions of genes and reactions (GR2 and R9).

Suppose that the kill matrix $\boldsymbol{K}$ and $\boldsymbol{H}$ contain all EM of a metabolic system, i.e. $\boldsymbol{G} = \boldsymbol{0}$. Then equation (12) allows to determine the maximum number of surviving EM. It is interesting to connect this result to the original formulation of the cMCS approach [16]. In their paper the authors define an intervention problem "by a set $\boldsymbol{T}$ of target modes and a set $\boldsymbol{D}$ of desired modes of which at least n must not be hit by a cMCS" [16]. Here, their $\boldsymbol{T}$ corresponds to our $\boldsymbol{K}$, while the row vectors of $\boldsymbol{H}$ will in general be a superset of $\boldsymbol{D}$. However, for any $\boldsymbol{T}$ equation (12) gives an upper bound to the preserved number n of desired EM, which is an important parameter in the cMCS-formulation.

Result
Realistic example

In analogy to [16] we validated our approach by predicting MCS for the efficient production of ethanol in *E. coli* using data presented by [12]. There, the authors used a small-scale metabolic model under anaerobic conditions, calculated all its 5,010 EM, optimized for the most efficient production of ethanol from glucose, and came up with a strain design where seven reactions were removed from the network. They found that only twelve EM contributed to the optimal design. All of them produced ethanol and four EM were also growth coupled. (The full model used by [12] is listed in the Additional file 1: Table S1.)

Using our algorithm we were able to design a cell with identical functional capabilities, but with fewer knockouts. In fact, the minimally necessary number of reaction deletions was six [consistent with identical findings in [16]]. In our simulation $\boldsymbol{G}$ consisted of the twelve optimal EM identified by [12], $\boldsymbol{H} = \boldsymbol{0}$, and $\boldsymbol{K}$ contained the remaining 4,998 EM. In less than 25 sec computation time we found 1,048 MCS of which 252 required exactly seven deletions. One of these MCS was the solution given by [12]. Again, our findings are identical with [16]. However, note that 71% of these 1,048 MCS, are not deletable due to missing annotations or are in principle undeletable. We used the gene-enzyme-reaction mapping as given by [12], who annotated only enzymatic reactions, but no transporters. Here, we consider all non-annotated reactions in the model of [12] as "undeletable". Most of these non-annotated reactions are transport reactions. Some of them may merely miss an annotation, and – in principle – could be deleted. Others however, are diffusion transporters and cannot be blocked. For simplicity, we do not distinguish between these two types and considered both as undelatable. In contrast however, we do consider uptake reactions as deletable – independent of any possible annotation – as these transporters are simply "deletable" by removing the substrate form the medium.

By using a weight function (one possible function is given in the Additional file 1: Table S2) our algorithm is able to predict biologically feasible deletions first. In

fact, the *in vivo* implementation of the smallest, fully annotated, biologically feasible MCS requires seven gene deletions. We found eight alternate MCS. In comparison, the experimentally implemented strain by [12] had eight knockouts.

To test the robustness of the alternate optimal solutions against variation in the weight vector, we randomly changed each weight in the range between ± 20% and repeated our calculation 1,000 times. Every time we found the same eight solutions with seven deletions. To further test the stability of our predictions, we incrementally changed each weight in such a way that after 150 steps all weights are one and thus recover the situation without weights (see the Additional file 1: Figure S1). Even with this procedure we find stable predictions over a wide range of different weights. (For details on the procedure and specific results we refer to the Additional file 1: section "Robustness of optimal solutions against variations in the weight vector" and Additional file 1: Figure S1.)

However, even with a weight function it is possible to fully enumerate all solutions.

To test wether our algorithm is able to handle larger system we repeated the analysis with the full model used by [12], that is, without restricting the model to glucose uptake under anaerobic conditions first. The complete model contained 429,276 EM – including the elementary futile cycle succinate dehydrogenase and its reverse reaction fumarate reductase (reactions R_TCA10 and R_TCA7 in the model). This cycle was disregarded in the following analysis.

Again, we used the same twelve EM (identified by Trinh *et. al.* [12] and defined above) as design criterion. (That is, the goal matrix **G** consisted of the twelve optimal EM, **H** = **0**, and **K** contained the remaining EM.) Without any weight function and additional constraints, at least eleven reaction deletions are required to reach the design goal. In total we found 55,488 MCS, 1.440 of which require the minimal number of eleven reaction deletions. (For the sake of completeness we listed the maximal number of MCS as function of deletions in the Additional file 1: Table S3.) Note, however, that these deletions are knockouts of reactions without regard to biological feasibility. In fact we found that none of those 1.440 MCS are fully annotated. Furthermore, only 27.7% of all 55,488 MCS are fully annotated, enzymatic reactions. In all other MCS at least one reaction was a transport reaction, for which genetic information was lacking. In order to calculate biologically feasible solutions first we included the weight function given in the Additional file 1: Table S2.

Using the weight function listed in the Additional file 1: Table S2, at least twelve reaction deletions were required to reach the design goal. Out of these twelve deletions five are uptake reactions (L-arabinose, D-galactose,

D-mannose, D-xylose, and oxygen). Removing these substrates from the growth medium recovers the initial model discussed above: anaerobic growth with glucose as the sole carbon source. We found eight equivalent solutions. Those solutions are exactly the solutions predicted for the anaerobic model above.

In their paper [12] the authors noted that six out of the twelve EM in the optimal design are inactive. Those six, inactive EM use the pyruvate dehydroganese complex (reaction R_GG13) which is down regulated under anaerobic conditions (reaction R_TRA11 = 0) [12,21]. Additionally the repression of the glyoxylate shunt (reaction R_GLB1) during growth on glucose (reaction R_GG1) [22,23] has not been considered in the above analysis. With our algorithm, however, it is possible to simply include these regulatory information in the form of two additional constraints, R_GG1 + R_GLB1 ≤ 1, and R_TRA11 + R_GG13 ≤ 1. Note however, an analysis which combines the regulatory information with the previous design goals (i.e. those twelve desired EM) does not yield a solution as the desired design and the regulatory constraints are inconsistent.

We repeated the analysis, included weights and regulatory constraints, and used the six potentially active EM as goal matrix. We predicted 13 deletions. Two of which (R_GLB1 and R_GG13) are, however, no deletions, but in fact the result of the regulation. As expected, we found that all MCS were characterized by a down-regulation of R_GLB1 and R_GG13.

Discussion

Elementary mode analysis has been identified as a promising tool for metabolic engineering. However, the analysis of millions or billions of EM still poses difficulties. Recently [16] introduced the concept of cMCS, which allows calculating all optimal metabolic engineering strategies. Here we showed that equivalent results can be obtained by simple integer programming.

We partitioned EM into three categories: goal modes, kill modes, and helper modes. The first group contains the desired functionality. All modes in this group will also be present in the final NMF. Kill modes, on the other hand, are all those modes which will definitely get deleted from the network. The third and final group collects all other modes, which may or may not be present in the final design. With respect to the design criterion this last group of modes neither contributes to nor counteracts the design goal. We then reformulated the problem of calculating MCS to generate NMF as a linear optimization problem. Our approach is very intuitive and structurally reminiscent to ordinary flux balance analysis. The matrix of EM replaces the stoichiometric matrix. Constraints are not set by the mass-balance but by design requirements. More specifically, binarized EM show up as constraints

on the admissible flux space. By optimizing the admissible flux space, maximal or minimal intervention strategies (with respect to the number of deletions) can be predicted.

In the case of optimal ethanol production we demonstrated that BLP is able to give identical results as compared to cMCS [16]. In fact, in a special case our method is formally equivalent to cMCS (see section "Definitions"). Moreover, by optimizing metabolic functionality BLP allows to calculate an upper bound for the maximum number of persevered EM, which is an important parameter in the cMCS-formulation. The two methods differ in that BLP uses a fixed set of desirable EM – the goal matrix G –, while in cMCS EM are chosen automatically form a pool of desirable EM, D, such that at least n modes survive. However, if the surviving modes are known an identical BLP problem can be set up.

The major advantage of our reformulation is its easy integration of (binary) transcriptional regulation. Regulatory information may simply be included as additional constraints. We have shown that our formulation allows a regulatory coupling between reactions and between genes and reactions alike. In Table 4 and in Table 3, we listed several examples for simple regulatory interactions. However, these expressions are easily expandable to more complex functions. The mapping between genes, proteins and reactions, as well as transcriptional regulation can be included as long as they are formulated as static boolean constraint. At least for well studied organisms like *E. coli* and *S. cerevisiae* curated transcriptional regulatory networks are readily available [24]. However, dynamic regulations or cyclic causalities pose immense difficulties and cannot be represented in our approach.

Additionally the BLP formulation offers more flexibility in the way solutions are predicted. By using weights in the objective function it is possible to account for experimental difficulties in the implementation of the strain. This allows to prioritize biologically feasible MCS over infeasible ones and – in contrast to other, optimized based methods [9] – does not effect the ability to calculate the complete set. Taking biological feasibility into account seems advantageous as in our example we have demonstrated that due to the lacking gene-enzyme-reaction mapping roughly 70% of the predicted solutions would require the deletion of at least one non-enzymatic reaction. Due to the combinatorial explosion of the number of EM [25], we expect that the percentage of unrealizable solutions is increasing further with augmenting system size. Obviously, sorting of solutions with respect to biological feasibility can be done in a separate post-processing step, too. However, in our implementation we get the sorting for "free", i.e. without any additional computational steps. At least in the case of *E. coli* we demonstrated that our predictions are robust against variations in the weights. In particular we found that our choice of weights

is very conservative and far from the limits detected in the robustness analysis (see the Additional file 1: Figure S1). It may be possible to integrate a weighting function in the algorithm presented by [16] as well. However, it has not been demonstrated yet.

Part of the flexibility of our approach is its ability to optimize metabolic functionality. This can be easily demonstrated in a simple example as illustrated in Figure 3. The network consists of three EM. Lets suppose that R1-R2-R3 is the only desirable EM (G = R1-R2-R3, H = R5-R4-R3, and K = R1-R6-R4-R3). A NMF can be easily generated by either knocking out either R4 or R6. The metabolic functionality of the resulting networks, however, differs significantly. By deleting R4 the only "surviving" mode is the desired goal mode. Thus the network is in fact a NMF, as no other functionality is available. On the other hand, by deleting R6 the network still has the desired properties, but retains additional functionality (conversion of C to B) without compromising the original design criterion. Alternatively we may define a kill-matrix K and calculate the resulting network of minimal or maximal functionality. BLP is able to predict solutions with the smallest/largest set of EM contributing to the metabolic functionality and distinguish between those two extremes without enumerating the full solution space. This feature therefore opens a way to include secondary objectives in the design process.

An integer programming problem sits at the heart of our algorithm. Integer programs are inherently difficult to solve [20]. Nevertheless efficient commercial and non-commercial solvers are available. Still the question remains if BLP is fit for solving even larger problems than the one presented. Even with current technologies a complete EMA can only be done for small-scale problems, typically involving about 100 reactions. These 100 reactions transform into 100 binary variables in the BLP problem. Their handling is easy [20]. On the other hand, the number of EM in metabolic networks explodes combinatorially with the system size [25], which translates into millions and even billions of constraints for BLP. These constraints are highly redundant and can be efficiently compressed using various preprocessing techniques typically already included in available solvers (or various preprocessing methods see [20,26]). For instance, the initial BLP problem to predict the smallest MCS in the full *E. coli* model [12] (see above) contained 429,276 constraints for 71 variables. After preprocessing we transformed the problem into 28 constraints for 34 variables, which dramatically improved the computational performance (data not shown). The compression is also beneficial in the context of the original formulation of cMCS. The problem may be set up as integer program first, followed by standard preprocessing. The reduced problem may then be solved by the adapted Berge algorithm presented by [16].
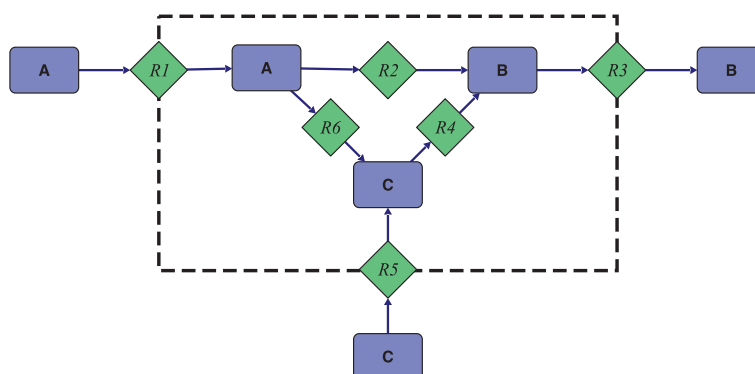
Figure 3 Illustrative toy network. Illustrative toy network containing the metabolites A to C, and the reactions R1 to R6. All reactions are irreversible. The area inside the dashed box indicates the "cell interior". The network consists of three EM: R1-R2-R3, R1-R6-R4-R3, and R5-R4-R3.

We tested various off-the-shelf software packages to solve the BLP problem. The implementation of our algorithm merely required setting up the input parameters for those solvers. We found that our approach is computationally modest and scalable. In fact, we were able to successfully repeat the analysis for the much larger core metabolic network of [24] with its 271 million EM on a standard personal computer. We used the single most efficient EM for the production of ethanol form glucose as design criterion (all other modes were killed). In 122 sec our algorithm found all 2,304 MCS with the minimum number of 26 deletions. (The total program runtime, which included reading all EM from disk and calculating the MCS, was 10 min 30 sec.) The problem here, and with cMCS in general, is not the handling of millions of EM (although data handling required 80% of the total runtime), but to calculate these modes in the first place [16]. However, promising results on efficiently enumerating the full set of EM have recently been published [27,28].

Conclusion

In summary, we have demonstrated an efficient and easy to implement method to rationally predict engineering strategies for the improvement of production hosts. Optimal pathways were identified using elementary mode analysis. Based on integer/binary programming we were then able to predict all minimal intervention strategies to design a strain with desirable metabolic capabilities. Our method is based on the concept of constrained minimal cut sets, but offers much more flexibility in the prediction of engineering targets, including most prominently the possibility of easily integrating gene regulation.

Methods

We used *efmtool* [28] to calculate the complete set of EM for a network and Gurobi Optimizer 5.0, http://www.gurobi.com/ for solving the BLP problem. *efmtool* is open

source and freely available; Gurobi offers a free academic license.

Additional file

Additional file 1: Supplementary material. A pdf containing all additional data, figures and tables.

References
1. Park JM, Kim TY, Lee SY: **Constraints-based genome-scale metabolic simulation for systems metabolic engineering.** *Biotechnol Adv* 2009, **27**(6):979–988. [http://www.sciencedirect.com/science/article/pii/S0734975009001141].
2. Orth JD, Thiele I, Palsson BO: **What is flux balance analysis?** *Nat Biotech* 2010, **28**(3):245–248. [http://dx.doi.org/10.1038/nbt.1614].
3. 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. [http://onlinelibrary.wiley.com/doi/10.1002/bit.10803/abstract].
4. Pharkya P, Maranas CD: **An optimization framework for identifying reaction activation/inhibition or elimination candidates for overproduction in microbial systems.** *Metab Eng* 2006, **8**:1–13. [http://www.sciencedirect.com/science/article/pii/S1096717605000704].
5. Patil KR, Rocha I, Förster J, Nielsen J: **Evolutionary programming as a platform for in silico metabolic engineering.** *BMC Bioinformatics* 2005,

6:308. [PMID: 16375763]. [http://www.ncbi.nlm.nih.gov/pubmed/16375763].

6. Pharkya P, Burgard AP, Maranas CD: **OptStrain: A computational framework for redesign of microbial production systems.** *Genome Res* 2004, **14**(11):2367–2376. [http://genome.cshlp.org/content/14/11/2367.abstract].

7. 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. [http://dx.doi.org/10.1371/journal.pcbi.1000744].

8. Trinh CT, Li J, Blanch HW, Clark DS: **Redesigning Escherichia coli Metabolism for Anaerobic Production of Isobutanol.** *Appl and Environ Microbiol* 2011, **77**(14):4894–4904.

9. Boghigian BA, Shi H, Lee K, Pfeifer BA: **Utilizing elementary mode analysis, pathway thermodynamics, and a genetic algorithm for metabolic flux determination and optimal metabolic network design.** *BMC Syst Biol* 2010, **4**:49. [http://www.biomedcentral.com/1752-0509/4/49].

10. Unrean P, Trinh CT, Srienc F: **Rational design and construction of an efficient E. coli for production of diapolycopendioic acid.** *Metab Eng* 2010, **12**(2):112–122. [http://www.sciencedirect.com/science/article/pii/S1096717609001050].

11. Trinh CT, Srienc F: **Metabolic Engineering of Escherichia coli for Efficient Conversion of Glycerol to Ethanol.** *Appl and Environ Microbiol* 2009, **75**(21):6696–6705. [http://aem.asm.org/content/75/21/6696.abstract].

12. Trinh CT, Unrean P, Srienc F: **Minimal Escherichia coli Cell for the Most Efficient Production of Ethanol from Hexoses and Pentoses.** *Appl Environ Microbiol* 2008, **74**(12):3634–3643. [http://aem.asm.org/content/74/12/3634.abstract].

13. Carlson R, Fell D, Srienc F: **Metabolic pathway analysis of a recombinant yeast for rational strain development.** *Biotechnol Bioeng* 2002, **79**(2):121–134. [http://onlinelibrary.wiley.com/doi/10.1002/bit.10305/abstract].

14. Schuster S, Fell DA, Dandekar T: **A general definition of metabolic pathways useful for systematic organization and analysis of complex metabolic networks.** *Nat Biotech* 2000, **18**(3):326–332. [http://dx.doi.org/10.1038/73786].

15. Schuster S, Dandekar T, Fell DA: **Detection of elementary flux modes in biochemical networks: a promising tool for pathway analysis and metabolic engineering.** *Trends in Biotechnol* 1999, **17**(2):53–60. [http://www.sciencedirect.com/science/article/pii/S0167779998012906].

16. Hädicke O, Klamt S: **Computing complex metabolic intervention strategies using constrained minimal cut sets.** *Metab Eng* 2011, **13**(2):204–213. [http://www.sciencedirect.com/science/article/pii/S1096717610001084].

17. Balas E, Jeroslow R: **Canonical cuts on the unit hypercube.** *SIAM J Appl Mathematics* 1972, **23**:61–69. [http://www.jstor.org/stable/2099623].

18. Tsai JF, M-H Lin YCH: **Finding multiple solutions to general integer linear programs.** *Eur J Operational Res* 2008, **184**(2):802–809. [http://dx.doi.org/10.1016/j.ejor.2006.11.024].

19. Covert MW, Knight EM, Reed JL, Herrgard MJ, Palsson BO: **Integrating high-throughput and computational data elucidates bacterial networks.** *Nature* 2004, **429**(6987):92–96. [http://dx.doi.org/10.1038/nature02456].

20. Chen D, Batson RG, Dang Y: *Applied Integer Programming: Modeling and Solution.* 1edition. Hoboken, New Jersey: John Wiley & Sons, Inc; 2010. ISBN:978-0-470-37306-4.

21. Alexeeva S, De Kort B, Sawers G, Hellingwerf KJ, De Mattos MJT: **Effects of Limited Aeration and of the ArcAB System on Intermediary Pyruvate Catabolism in Escherichia Coli.** *J Bacteriol* 2000, **182**(17):4934–4940. [http://jb.asm.org/content/182/17/4934].

22. Kornberg HL: **The role and control of the glyoxylate cycle in Escherichia coli.** *Biochem J* 1966, **99**:1–11. [PMID: 5337756 PMCID: 1264949].

23. Kornberg HL: **The role and maintenance of the tricarboxylic acid cycle in Escherichia coli.** *Biochem Soc Symp* 1970, **30**:155–171. [PMID: 4322317]. [http://www.ncbi.nlm.nih.gov/pubmed/4322317].

24. Orth JD, Fleming RMT, Palsson BO: **10.2.1 - Reconstruction and use of microbial metabolic networks: the core Escherichia coli metabolic model as an educational guide.** In *EcoSal - Escherichia coli and Salmonella Cellular and Molecular Biology.* Edited by Karp P D. 10.2.1 Washington DC: ASM Press; 2010.

25. Klamt S, Stelling J: **Combinatorial Complexity of Pathway Analysis in Metabolic Networks.** *Mol Biol Reports* 2002, **29**:233–236. [http://www.springerlink.com/content/x2m6215394074117/abstract/].

26. Haus U, Klamt S, Stephen T: **Computing Knock-Out Strategies in Metabolic Networks.** *J Comput Biol* 2008, **15**(3):259–268. [http://www.liebertonline.com/doi/abs/10.1089/cmb.2007.0229].

27. Jevremović D, Trinh CT, Srienc F, Sosa CP, Boley D: **Parallelization of Nullspace Algorithm for the computation of metabolic pathways.** *Parallel Comput* 2011, **37**(6–7):261–278. [http://www.sciencedirect.com/science/article/pii/S0167819111000378].

28. Terzer M, Stelling J: **Large-scale computation of elementary flux modes with bit pattern trees.** *Bioinformatics* 2008, **24**(19):2229–2235. [http://bioinformatics.oxfordjournals.org/content/24/19/2229.abstract].