Proceedings of the 9th International Symposium on
Dynamics and Control of Process Systems (DYCOPS 2010),
Leuven, Belgium, July 5-7, 2010
Mayuresh Kothare, Moses Tade, Alain Vande Wouwer, Ilse Smets (Eds.)

Rapid design of system-wide metabolic network modifications using iterative linear programming

Laurence Yang * **William R. Cluett** *
Radhakrishnan Mahadevan *,**

* *Department of Chemical Engineering & Applied Chemistry, University of Toronto, Toronto, ON M5S3E5, Canada.*
** *Institute of Biomaterials and Biomedical Engineering, University of Toronto, Toronto, ON M5S3E5, Canada.*
(email: laurence.yang@utoronto.ca, will.cluett@utoronto.ca, krishna.mahadevan@utoronto.ca)

Abstract: Computationally-aided metabolic engineering is an important, complementary strategy to combinatorial strain design for enhanced biochemical production by microbes. Bilevel optimization problems have been formulated for optimal strain design via reaction removal, activation, and inhibition. Deterministic global optimization of the resulting mixed integer linear programs (MILPs) requires extensive computational effort, especially for genome-scale models of metabolism. Improving the computational efficiency of such algorithms is an ongoing challenge. Here, we present Enhancing Metabolism with Iterative Linear Optimization (EMILiO)–a novel bilevel optimization-based algorithm that includes all possible flux modifications and is solved with remarkable computational efficiency via iterative linear programming. The resulting solution is recursively pruned to generate alternate, parsimonious strain designs with maximal biochemical production rates. We demonstrate our algorithm for succinate production using the *i*AF1260 genome-scale model of *Escherichia coli* metabolism. Our algorithm identifies aerobic succinate-producing strains with increased glyoxylate shunt activity, which is consistent with experiments in the literature. We also identified novel strain design strategies that may have implications for the control of industrial bioreactors to maximize succinate production.

Keywords: mathematical programming, biotechnology, integer programming, iterative linear programming, metabolic engineering, complementarity constraints, successive linear programming, flux balance analysis, succinate production

1. INTRODUCTION

Microbial cell factories are becoming increasingly important for the sustainable production of chemicals and fuels. Improving the efficiency of production from microbes relies, in large part, on systematically engineering their metabolism through genetic modifications. With flux balance analysis (FBA) (Edwards et al., 2002), the reaction fluxes of metabolic networks are simulated at the genome-scale as a linear program (LP). FBA has been used to accurately predict cell physiology by integrating multiple types of high-throughput data, especially for industrially important microorganisms (Mahadevan et al., 2005).

Consequently, a number of computational algorithms have been developed to identify network manipulation strategies while predicting their system-wide effects. Burgard et al. (2003) developed OptKnock to identify a set of gene deletions that couples biochemical formation with the maximization of growth rate. Hence, by culturing the mutant strains under environments that impose a selective pressure for maximal growth rate, product formation is also maximized–this strategy was confirmed experimentally by Fong et al. (2005). Similarly, OptReg (Pharkya and Maranas, 2006) identifies target reactions for activation or inhibition, in addition to removal, to maximize biochemical production. These additional modifications are implemented using binary variables that enforce flux above or below a pre-defined deviation from a defined reference flux.

Globally optimal solutions to OptKnock and OptReg can be found using deterministic mixed integer linear program (MILP) solvers; however, due to the large size of genome-scale models and a combinatorial space involving hundreds of binary variables, prohibitively long computational times can be required to obtain global solutions. Hence, in practice, the maximum number of network modifications is restricted to a small number (e.g., three or four).

Recently, Lun et al. (2009) developed Genetic Design through Local Search (GDLS) to overcome the computational limitations of obtaining a global optimum to OptKnock. GDLS finds a local optimum by iteratively solving the MILP with a cap on the maximum number of changes to the binary variable set, relative to the solution from the previous iteration. This cap, termed the neighborhood

⋆ The authors gratefully acknowledge the Natural Sciences and Engineering Research Council of Canada for financial support.

size, reduces the combinatorial space of each MILP. The deterministic solution of MILPs at every iteration ensures monotonic convergence to a local optimum. GDLS is capable of finding complex microbial strain designs with dozens of network modifications in a fraction of the time required to obtain designs of similar scope using OptKnock. GDLS has been shown to predict complex designs with higher *in silico* production rates than similar algorithms based on evolutionary algorithms (Lun et al., 2009).

Despite these improvements, GDLS can still require significant computational effort due to iterative solutions of MILPs. For example, in (Lun et al., 2009), the authors report a strain design for succinate production that required over 4.5 hours ($k = 3, M = 1$ in Supplementary Table 2). Surprisingly, this design predicted less *in silico* succinate production than another run of GDLS using a smaller neighborhood size ($k = 1, M = 1$ in Supplementary Table 2). Hence, additional effort might be required to optimally tune the parameters of GDLS. In addition, GDLS has not been applied to the more difficult OptReg problem, where activation or inhibition of fluxes is included in the design.

In this paper, we develop a novel computational algorithm, termed *E*nhancing *M*etabolism with *I*terative *Li*near *O*ptimization (EMILiO), to identify strain design strategies that include reaction removal, activation, inhibition, as well as optimal flux direction of reversible reactions. Unlike OptReg, the algorithm does not require prior determination of a reference flux. Despite the comprehensive scope of the strain design, our algorithm requires minimal computational effort. This is because, unlike all of the aforementioned algorithms, we have formulated the optimal strain design problem as a mathematical program with complementarity constraints (MPCC) (similar to Yang et al. (2008)). In this work, we have efficiently solved the MPCC using iterative linear programming (ILP) (Bullard and Biegler, 1991).

The rest of the paper is organized as follows: we present the necessary frameworks for modeling metabolism in Section 2, describe our algorithm in Section 3, describe the computational experiments in Section 4, report and discuss our results in Section 5, and provide conclusions of the paper in Section 6.

2. PRELIMINARIES

Cell metabolism is modeled as a network consisting of hundreds of biochemical species, or metabolites, that are interconverted via enzyme-catalyzed reactions. The distribution of reaction fluxes throughout the network can be simulated using FBA. In FBA, the reaction network stoichiometry is defined in a matrix, S. Because the response times of metabolic flux distributions are often orders of magnitude faster than external perturbations to the cell, we assume pseudo-steady state of metabolite concentrations and reaction fluxes as follows:

$$Sv = \frac{dx}{dt} = 0, \tag{1}$$

$$v^L \leq v \leq v^U, \tag{2}$$

where $v \in \mathbb{R}^N$ is the vector of fluxes, $x \in \mathbb{R}^M$ is the vector of metabolite concentrations, $v^L \in \mathbb{R}^N$ and $v^U \in \mathbb{R}^N$ are the vectors of minimum and maximum fluxes, respectively. $S \in \mathbb{R}^{M \times N}$ is the matrix defining network stoichiometry with M rows corresponding to metabolites and N columns corresponding to fluxes.

Under environments with selective pressure for maximal growth rate, we can simulate the flux distribution by solving the following linear program (LP):

$$\begin{aligned} \max_{v} \quad & c^T v = v_{bio} \\ \text{s.t.} \quad & Sv = 0 \\ & v^L \leq v \leq v^U, \end{aligned} \tag{FBA}$$

where $c \in \mathbb{R}^N$ is the objective vector to maximize the growth rate, v_{bio}.

The rank, r, of S is less than M; hence, we can separate the free and pivot variables in the reduced row echelon form of S and formulate a reduced FBA problem as below:

$$\max_{v} \; c^T \cdot T v^f = v_{bio} \tag{3a}$$

$$\text{s.t. } v^L \leq T v^f \leq v^U, \tag{3b}$$

where $v^f \in \mathbb{R}^{N-r}$ are the free variables and $T \in \mathbb{R}^{N \times (N-r)}$ is defined such that $v = Tv^f$.

3. METHODS

Here, we develop a computational algorithm to maximize biochemical production via reaction removal, activation, inhibition, and restriction of flux direction for reversible reactions. The algorithm is formulated as the following bilevel optimization problem with the continuous flux bounds as decision variables:

$$\begin{aligned} \max_{v^L, v^U} \; & c_p^T \cdot T v^f \\ \text{s.t.} \quad & \max_{v^f} \; c^T \cdot T v^f - \epsilon \cdot c_p^T \cdot T v^f \\ & \text{s.t. } v^L \leq T v^f \leq v^U \\ & v_{bio} \geq v_{bio}^{min}, \end{aligned} \tag{4}$$

where v_{bio}^{min} is the minimum required growth rate, and the inner optimization is the reduced FBA formulation (3) with the additional objective of minimizing production rate. Hence, our algorithm identifies manipulation strategies having a high minimal production rate, when growth rate is optimal. Here, $\epsilon = 0.001$ is chosen so that the maximum growth rate is not affected by minimization of production. Using the KKT conditions, this bilevel optimization problem can be reformulated as a single-level mathematical program with complementarity constraints (MPCC) (Yang et al., 2008) as follows:

$$\max_{x} \quad c_p^T \cdot T v^f \tag{5a}$$

$$w_i^L \mu_i^L + w_i^U \mu_i^U = 0, \qquad i = 1, \ldots, N \tag{5b}$$

$$T v^f + \mu^U = v^U \tag{5c}$$

$$T v^f - \mu^L = v^L \tag{5d}$$

$$w^U T - w^L T = c^T \cdot T - \epsilon \cdot c_p^T \cdot T \tag{5e}$$

$$v_{bio} \geq v_{bio}^{min} \tag{5f}$$

$$w^L, w^U, \mu^L, \mu^U \geq 0 \tag{5g}$$

where $\mu^L \in \mathbb{R}^N$ and $\mu^U \in \mathbb{R}^N$ are slack variables for the lower and upper bounds, respectively, and $x = [v^f, v^U, v^L, \mu^U, \mu^L, w^U, w^L]^T$. The reduced FBA formulation has removed the need to include dual variables for $Sv = 0$, resulting in a smaller problem size.

3.1 Iterative Linear Programming for Strain Design

In Yang et al. (2008), the authors solved a similar MPCC by expressing the bilinear constraints (5b) as a penalty function and solving the resulting NLP using off-the-shelf NLP solvers. Here, we solve the above MPCC by formulating an iterative linear program (ILP).

Iterative linear programming was developed to solve a general nonlinear system of equations subject to nonlinear inequality constraints and variable bounds (Bullard and Biegler, 1991). The ILP converges to a feasible solution by iteratively generating search directions based on local linearization of the nonlinear equations and inequalities. In our algorithm, an ILP is formulated to satisfy the bilinear constraints (5b), while also maximizing product formation. Thus, at each iteration, k, we move the current solution, x^k, which violates the bilinear constraints but satisfies (5c)–(5g), by computing an optimal direction, u, and updating the solution, $x^{k+1} = x^k + u$.

For simplicity of notation, we define $e \in \mathbb{R}^{2N}$ and $f \in \mathbb{R}^{2N}$ such that $e^T x = [w^U, w^L]^T$ and $f^T x = [\mu^U, \mu^L]^T$. We, furthermore, define $g_i(x^k) = e_i^T x^k f_i^T x^k$. The bilinear constraints (5b) at iteration $k+1$ are expressed as $g_i(x^k + u) = 0$. We now construct a merit function, $Z(x^k)$, as in Bullard and Biegler (1991) but with the added objective of maximizing production rate:

$$Z(x^k) = \sum_{i=1}^{2N} g_i(x^k) - K_p \cdot c_p^T \cdot T v^f, \tag{6}$$

where K_p is a constant that controls the emphasis placed on maximizing production rate, relative to minimizing violation of the bilinear constraints. All results were obtained with $K_p = 1000$, but a dynamic K_p^k is also possible.

We can linearize $g_i(x^k + u)$ about x^k as $g_i(x^k) + \nabla g(x^k)u$, where $\nabla g(x^k)u = e_i^T x^k f_i^T u + f_i^T x^k e_i^T u$ is the directional derivative of $g(x^k)$ about x^k, in the direction u. We thus formulate the following LP to compute the optimal direction to minimize $Z(x^{k+1}) = Z(x^k + u)$:

$$\begin{aligned}
\min_{u,s} \quad & \sum_{i=1}^{N} s_i - K_p \cdot c_p^T \cdot T \Delta v^f && \text{(7a)}\\
\text{s.t.} \quad & g_i(x^k) + \nabla g_i(x^k) u \le s_i && \text{(7b)}\\
& T(v^f + \Delta v^f) + (\mu^U + \Delta \mu^U) = (v^U + \Delta v^U) && \text{(7c)}\\
& T(v^f + \Delta v^f) - (\mu^L + \Delta \mu^L) = (v^L + \Delta v^L) && \text{(7d)}\\
& (w^U + \Delta w^U)T - (w^L + \Delta w^L)T = c^T \cdot T - \epsilon \cdot c_p^T \cdot T && \text{(7e)}\\
& v_{bio} + \Delta v_{bio} \ge v_{bio}^{min} && \text{(7f)}\\
& w^L + \Delta w^L \ge 0 && \text{(7g)}\\
& w^U + \Delta w^U \ge 0 && \text{(7h)}\\
& \mu^L + \Delta \mu^L \ge 0 && \text{(7i)}\\
& \mu^U + \Delta \mu^U \ge 0 && \text{(7j)}\\
& s \ge 0, && \text{(7k)}
\end{aligned}$$

where $u = [\Delta v^f, \Delta v^U, \Delta v^L, \Delta \mu^U, \Delta \mu^L, \Delta w^U, \Delta w^L]^T = x^{k+1} - x^k$ is the direction vector, and $s \in \mathbb{R}^N$ are auxiliary variables used to minimize the bilinear constraints to 0.

Upon calculating the optimal direction, u^*, a line search is performed at each iteration to determine the maximum step size that ensures monotonic improvement of the objective value, $Z(x^k)$. For general nonlinear constraints, Bullard and Biegler (1991) propose a monotonic, Armijo-type line search to determine the maximum step size. Here, we can compute the maximum step size exactly because the directional derivative of our bilinear constraints results in a quadratic equation in terms of the step size. Hence, to ensure that $Z(x^k + \lambda u^*) \le Z(x^k)$ for $\lambda \ge 0$, we determine the maximum step size,

$$\lambda_{max} = \frac{K_p \cdot c_p^T \cdot T \Delta v^f - \sum_{i=1}^{2N} (e_i^T x^k f_i^T u^* + f_i^T x^k e_i^T u^*)}{\sum_{i=1}^{2N} e_i^T u^* f_i^T u^*}, \tag{8}$$

for $\sum_i e_i^T u^* f_i^T u^* > 0$. The actual step size is set to

$$\lambda = \min(1, \max(0, \lambda_{max})). \tag{9}$$

The solution is then updated as $x^{k+1} = x^k + \lambda u^*$. The ILP converges when $\lambda < StepTol = 10^{-6}$, indicating that no further improvement of the objective function is possible. The line search is critical to ensure convergence of the ILP to a local optimum from arbitrary starting points, as shown by Bullard and Biegler (1991).

3.2 Pruning the Design Using LP

The solution of the ILP in Section 3.1 generates modified lower and upper bounds $\tilde{v}^L$ and $\tilde{v}^U$. We define the design sets, $DesignL$ and $DesignU$ as the N_L lower and N_U upper bounds that are different from the original bounds and whose corresponding dual variables are strictly positive. Due to network redundancy, many of these constraints may not be active, simultaneously. Hence, smaller subsets of active constraints may exist. We extract such subsets by recursively solving the following LP:

$$\begin{aligned}
\min_{v} \quad & c_p^T v && \text{(LPR)}\\
\text{s.t.} \quad & Sv = 0 &&\\
& \tilde{v}_i^L \le v_i, && \forall i \in DesignL\\
& v_i \le \tilde{v}_i^U, && \forall i \in DesignU\\
& v_i^L \le v_i, && \forall i \in \{1, \ldots, N\} \text{ and } i \notin DesignL\\
& v_i \le v_i^U && \forall i \in \{1, \ldots, N\} \text{ and } i \notin DesignU\\
& v_{bio} \ge v_{bio}^{min}. &&
\end{aligned}$$

The solution to (LPR) is the minimum production rate, v_{prod}^*, subject to the modified bounds and minimal growth rate. We first determine if this minimum production rate is acceptable, say $v_{prod}^* \ge 0.5 \times v_{prod}^{max}$. We identify the set of active bound constraints and define it as a subset strain design. We remove these active constraints from $DesignL$ and $DesignU$ and solve (LPR) again, with the remaining modified bounds. We then define another strain design if the resulting production rate is still acceptable. We recursively apply this procedure to all strain designs and their subset strain designs. We terminate the procedure when no strain design yields a subset design that is smaller in size, or if all of these subset designs exhibit lower production rate than the defined tolerance of $0.5 \times v_{prod}^{max}$.

3.3 Minimal and Alternate Optimal Designs Using MILP

The recursive pruning phase in Section 3.2 may produce alternate strain designs that are more parsimonious than

the single initial set generated in Section 3.1. The LP in this pruning stage, however, has not been formulated to generate the strain design with the minimal number of modifications. We thus formulate a final processing phase as an MILP, with binary variables, $y^L \in \mathbb{Z}^{N_L}$ and $y^U \in \mathbb{Z}^{N_U}$, to identify the minimal set of reaction modifications to achieve a desired production rate of v_p^{min} as follows:

$$\begin{aligned}
\min_{y^L,y^U}\ & \sum_{i=1}^{N_L} y_i^L + \sum_{i=1}^{N_U} y_i^U \\
\text{s.t. } & \max_v\ c_{bio}^T v - \epsilon \cdot c_p^T v \\
& \text{s.t. } Sv = 0 \\
& \quad v^L \le v \le v^U \\
& \quad \tilde{v}_i^L y_i^L + v_{DL,i}^L (1 - y_i^L) \le v_{DL,i},\ i = 1,\dots,N_L \\
& \quad v_{DU,i} \le \tilde{v}_i^U y_i^U + v_{DU,i}^U (1 - y_i^U),\ i = 1,\dots,N_U \\
& c_p^T v \ge v_p^{min} \\
& y_i^L \in \{0,1\}, \qquad i = 1,\dots,N_L \\
& y_i^U \in \{0,1\}, \qquad i = 1,\dots,N_U,
\end{aligned} \tag{10}$$

where $v_{DL} = \{v_i : \forall i \in DesignL\}$, $v_{DU} = \{v_i : \forall i \in DesignU\}$, $v_{DL}^L = \{v_i^L : \forall i \in DesignL\}$, and $v_{DU}^U = \{v_i^U : \forall i \in DesignU\}$. This bilevel optimization problem is reformulated to a single level MILP as follows:

$$\begin{aligned}
\min_{\substack{v, w^S, \\ w^L, w^U, \\ \eta^L, \eta^U, \\ y^L, y^U}}\ & \sum_{i=1}^{N_L} y_i^L + \sum_{i=1}^{N_U} y_i^U \\
\text{s.t. } & Sv = 0 \\
& v^L \le v \le v^U \\
& \tilde{v}_i^L y_i^L + v_{DL,i}^L (1 - y_i^L) \le v_{DL,i},\ i = 1,\dots,N_L \\
& v_{DU,i} \le \tilde{v}_i^U y_i^U + v_{DU,i}^U (1 - y_i^U),\ i = 1,\dots,N_U \\
& (w^S)^T S + w^L - w^U + \eta^L - \eta^U = c_{bio}^T - \epsilon \cdot c_p^T \\
& \sum_{i=1}^{N_L} \eta_i^L \tilde{v}_i^L + \sum_{i=1}^{N} w_i^L v_i^L - \\
& \quad \sum_{i=1}^{N_U} \eta_i^U \tilde{v}_i^U - \sum_{i=1}^{N} w_i^U v_i^U - c_{bio}^T v + \epsilon \cdot c_p^T v = 0 \\
& 0 \le \eta_i^L \le K y_i^L, \qquad i = 1,\dots,N_L \\
& 0 \le \eta_i^U \le K y_i^U, \qquad i = 1,\dots,N_U \\
& 0 \le w_{DL,i}^L \le K(1 - y_i^L), \qquad i = 1,\dots,N_L \\
& 0 \le w_{DU,i}^U \le K(1 - y_i^U), \qquad i = 1,\dots,N_U \\
& c_p^T v \ge v_p^{min} \\
& w^L, w^U, w_{DL}^L, w_{DU}^U \ge 0 \\
& y_i^L \in \{0,1\}, \qquad i = 1,\dots,N_L \\
& y_i^U \in \{0,1\}, \qquad i = 1,\dots,N_U,
\end{aligned} \tag{11}$$

where $w^L \in \mathbb{R}^N$ and $w^U \in \mathbb{R}^N$ are dual variables for lower and upper bounds, respectively, $w_{DL}^L = \{w_i^L : \forall i \in DesignL\}$, $w_{DU}^U = \{w_i^U : \forall i \in DesignU\}$, $\eta^L \in \mathbb{R}^{N_L}$ and $\eta^U \in \mathbb{R}^{N_U}$ are dual variables for the modified lower and upper bounds, respectively, and $K = 100$. The combinatorial space of this MILP is much smaller than attempting to solve OptKnock because we limit modifications to only those included in each strain design generated in Section 3.2. With this MILP formulation, we can also identify alternate optimal strain designs via integer cuts.

4. *IN SILICO* EXPERIMENTS

We implemented our algorithm to design succinate–producing strains using the *i*AF1260 (Feist et al., 2007) genome-scale model of *E. coli* metabolism. We performed two separate runs of our algorithm, as described in Table 1: Search II includes all possible modifications, while Search I excludes the modification of flux directions for reversible reactions as this requires system-wide control of metabolite concentrations to alter the thermodynamic feasibility of reaction directions.

At each iteration of the ILP, an LP with 13,089 variables and 6,813 equality or inequality constraints was solved. The MILPs in the final pruning stage involved 5,964 variables, 2,930 equality or inequality constraints, and 18 (Search II) to 102 (Search I) binary variables. For comparison, the MILP corresponding to OptKnock would involve >1,500 binary variables. At each stage of the algorithm, we verified the strain designs by solving (FBA) with the addition of the designed bounds.

The "biomass_iAF1260_core" reaction in the *i*AF1260 model was used to simulate cell growth. All simulations were run with maximum glucose and oxygen uptake rates of 10 and 18.5 mmol/gDW/hr, respectively, and a minimum growth rate of 0.05 h^{-1}. We computed the maximum succinate production rate, $v_{prod}^{max} = 15.83$ mmol/gDW/hr by solving (FBA), except with the objective function, $\max c_p^T v$, subject to the minimal growth rate constraint. During the LP- and MILP-based design pruning, the minimum acceptable succinate production rate was set to 50% of v_{prod}^{max}. All code was implemented in MATLAB (The Mathworks, Inc., Natick, MA). CPLEX 12.1 was used to solve the LPs and MILPs using the MATLAB connector from IBM ILOG. All simulations were run on Intel Xeon 3.2 GHz processors with up to eight available CPUs.

5. RESULTS AND DISCUSSION

Our algorithm identified comprehensive strain designs, including a global optimum (100% v_{prod}^{max}) within minutes. The resulting strain designs included modifications consistent with previous work in the literature. Both Search I and II yielded a single strain design each. These strains exhibited aerobic succinate production, which has been shown to be important for overcoming bottlenecks associated with anaerobic fermentation strains (Lin et al., 2005). Both strains exhibited increased glyoxylate shunt activity via ICL (Search I) and MALS (Search II) activation (Figure 1). This strategy agrees with the previously observed increase in glyoxylate shunt activity in efficient succinate-producing strains (Lin et al., 2005). Total CPU time for all phases of the algorithm was ∼4 minutes (Table 2). The final MILP phase accounted for ∼50% of the total CPU time. This phase reduced the strain designs by 93 and 9 modifications for Search I and II, respectively. Alternate optima may exist further within the subsets generated by the LP-based pruning, especially for Search I.

An intriguing strategy involves limiting the maximum oxygen uptake rate, which is suggested in Search I via O2tpp limitation (Figure 1). To investigate the basis for

Table 1. Flux modifications and their physical implementation strategies

Strategy	Description	Implementation	Strategy Available?	
			Search I	Search II
$v^L = 0 = \tilde{v}^U < v^U$	Knockout irreversible (forward) reaction	Gene deletion	Yes	Yes
$v^L \leq \tilde{v}^L = 0 = v^U$	Knockout irreversible (reverse) reaction	Gene deletion	Yes	Yes
$0 \leq v \leq \tilde{v}^U < v^U$	Reduce forward flux	Down-regulation	Yes	Yes
$v^L < \tilde{v}^L \leq v \leq 0$	Reduce reverse flux	Down-regulation	Yes	Yes
$v^L \leq v \leq \tilde{v}^U < v^U \leq 0$	Increased reverse flux	Up-regulation	Yes	Yes
$0 \leq v^L < \tilde{v}^L \leq v \leq v^U$	Increased forward flux	Up-regulation	Yes	Yes
$v^L \leq v \leq \tilde{v}^U < 0 \leq v^U$	Forced reverse flux	Concentration ratios*	No	Yes
$v^L \leq 0 < \tilde{v}^L \leq v \leq v^U$	Forced forward flux	Concentration ratios*	No	Yes

* Requires system-wide control of metabolite concentration ratios to affect thermodynamic feasibility of reaction directions.
v denotes flux.
v^L and v^U denote wild-type lower and upper bounds, respectively.
$\tilde{v}^L$ and $\tilde{v}^U$ denote modified lower and upper bounds, respectively.

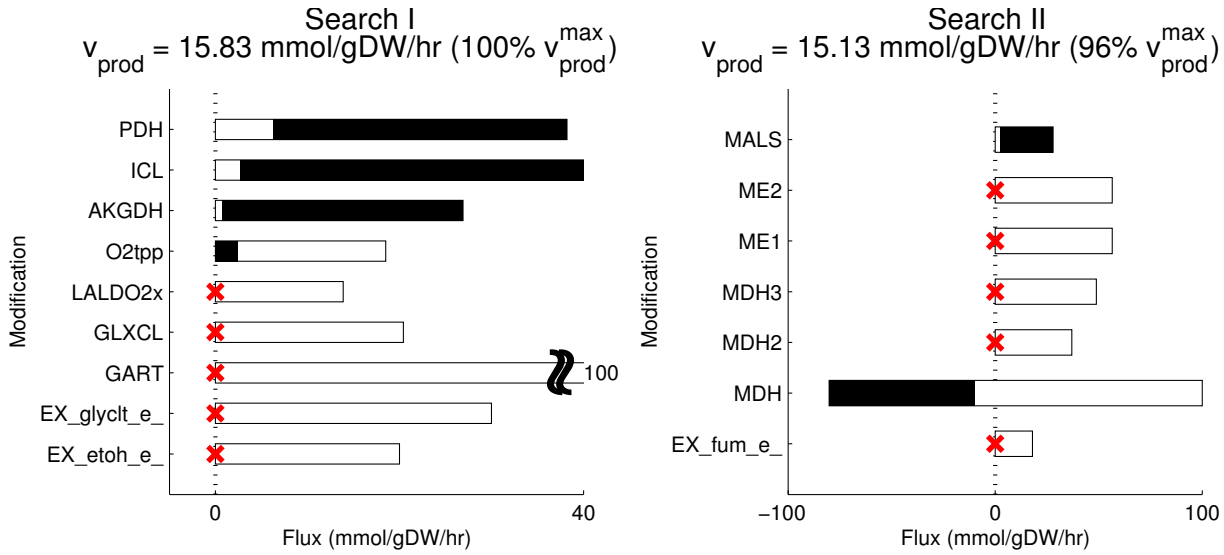

Fig. 1. Succinate production rates and the modifications proposed by our algorithm via Search I and II. The dark area corresponding to each reaction indicates the range of feasible flux, due to modification of that reaction. A cross indicates removal of the corresponding reaction. The thin-lined boxes indicate the range of feasible flux, prior to any reaction modifications, calculated using FVA (Mahadevan and Schilling, 2003). (AKGDH: 2 Oxogluterate dehydrogenase, ICL: Isocitrate lyase, GLXCL: Glyoxalate carboligase, GART: GAR transformylase T, LALDO2x: D-lactaldehyde-NAD-1 oxidoreductase, MALS: Malate synthase, ME1: Malic enzyme (NAD), ME2: Malic enzyme (NADP), MDH: Malate dehydrogenase, MDH2: Malate dehydrogenase (ubiquinone 8), MDH3: Malate dehydrogenase (menaquinone 8), PDH: Pyruvate dehydrogenase, EX_fum_e: Fumarate exchange, EX_glyclt_e: Glycolate exchange, EX_etoh_e: Ethanol exchange, O2tpp: Oxygen transport via diffusion (periplasm)).

Table 2. Process of generating and pruning designs using the *i*AF1260 genome-scale model of *E. coli* metabolism via Search I and II. Combined CPU times of both search strategies are shown.

Procedure	Number of reaction modifications [Lower bounds, Upper bounds]		CPU time (sec)
ILP	[487, 1193]	[391, 1122]	96.48
↓	↓	↓	
LP-pruning	[17, 85]	[4, 14]	18.75
↓	↓	↓	
MILP-pruning	[3, 6]	[1, 8]	116.58
Total	Search I	Search II	231.81

this strategy, we generated an additional strain by removing only the oxygen limitation strategy. We then simulated the maximum and minimum possible succinate production of the original and new strain with the minimum growth rate constraint via flux variability analysis (FVA) (Mahadevan and Schilling, 2003). We found that removing the oxygen limitation strategy did not change the maximum possible succinate production; however, the minimum succinate production decreased from 15.83 mmol/gDW/hr^{-1} (100% v_{prod}^{max}) to zero. Hence, for the strain design identified in Search I, oxygen uptake limitation is necessary to prevent low succinate production when the cellular objective is maximization of growth rate. Therefore, industrial succinate producing strains with modification strategies similar to those identified by our algorithm via Search I might achieve the greatest production rates when oxygen levels are optimally controlled in the bioreactors.

Search II allowed more degrees of freedom than Search I as we allowed the algorithm to determine the direction of reversible reactions (Table 1). Initially, our algorithm did identify a strain producing succinate at the globally opti-

mal rate. This strategy involved one lower and eight upper flux bound modifications. However, upon inspection of these strategies, we found that two of these reactions were forming a thermodynamically infeasible cycle to generate ATP. Once we removed these two modifications from the identified strain design, succinate production decreased slightly from 15.83 (100%) to 15.13 (96%) mmol/gDW/hr when growth rate was maximized (Figure 1). Hence, although Search II explored additional reaction modifications unavailable in Search I, not all modification strategies were physically implmentable. If such infeasibilities were removed *a priori*, Search II may have also yielded strain designs having the maximum succinate production rate. We note that some of these cycles and infeasible reaction directions can be eliminated by incorporating thermodynamic constraints in the model (Henry et al., 2007).

For Search II, we performed several iterations of our algorithm, each time assessing the physiological feasibility of each strain design. Because both pruning stages (recursive LP- and MILP-based) significantly reduced the number of modification strategies, the effort for such manual curation was minimal. As each complete run of our algorihm requires little computational effort, recursive strain design and curation of design strategies becomes feasible even for less characterized organisms, as well as for larger models such as the metabolism of microbial communities.

6. CONCLUSIONS

We have developed a computational algorithm to design strains for biochemical overproduction through reaction removal, activation, inhibition, and modification of flux direction. The bilevel optimization problem was reformulated into a single-level MPCC as in Yang et al. (2008), and was solved by iterative linear programming (Bullard and Biegler, 1991). A large lumped set of reaction modifications producing 100% of the maximum succinate production rate was identified using the ILP in ~ 1.5 minutes for the latest genome-scale model of *E. coli* metabolism. Parsimonious strain designs were identified using recursive LP-based and MILP-based pruning stages. The algorithm identified a strain design having 100% of the maximum succinate production, even when the modification of flux directions was omitted from the available design strategies (Figure 1: Search I).

Of the three stages of the algorithm, the MILP-based pruning stage accounted for half of the total computational effort (Table 2). The computational effort of the MILP increases when strain designs with larger numbers of reaction modifications are identified by the recursive LP-based pruning stage. In the future, we hope to improve the efficiency of the pruning stages. For example, we can formulate an ILP for each solution of the LP-based pruning step, with the objective of minimizing the number of active constraints while maintaining the desired production rate. The ability of our algorithm to identify maximal production strains owes largely to its ability to fine-tune reaction rates by directly modifying flux bounds. In practice, such fine-tuning of reactions might not be possible. In the future, we can modify the algorithm to reflect this physical limitation by imposing constraints on the maximum modification of the flux bounds. Additionally, we can more closely model genetic manipulations by using the gene-protein-reaction mappings, as in Lun et al. (2009). In the future, the three stages of the algorithm may be reformulated into one MINLP, which may facilitate extensions such as thermodynamic constraints (Henry et al., 2007). The remarkable computational efficiency of the ILP method for solving bilevel optimization problems with genome-scale models will undoubtedly open new doors for computationally-aided metabolic engineering.

REFERENCES

Bullard, L.G. and Biegler, L.T. (1991). Iterative linear programming strategies for constrained simulation. *Computers and Chemical Engineering*, 15, 239–254.

Burgard, A.P., Pharkya, P., and Maranas, C.D. (2003). OptKnock: A bilevel programming framework for identifying gene knockout strategies for microbial strain optimization. *Biotechnology and Bioengineering*, 84, 647–657.

Edwards, J., Covert, M., and Palsson, B. (2002). Metabolic modelling of microbes: the flux-balance approach. *Environmental Microbiology*, 4, 133–140.

Feist, A.M., Henry, C.S., Reed, J.L., Krummenacker, M., Joyce, A.R., Karp, P.D., Broadbelt, L.J., Hatzimanikatis, V., and Palsson, B.O. (2007). A genome-scale metabolic reconstruction for *Escherichia coli* K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information. *Molecular Systems Biology*, 3, 121.

Fong, S.S., Burgard, A.P., Herring, C.D., Knight, E.M., Blattner, F.R., Maranas, C.D., and Palsson, B.O. (2005). In silico design and adaptive evolution of *Escherichia coli* for production of lactic acid. *Biotechnology and Bioengineering*, 91, 643–648.

Henry, C.S., Broadbelt, L.J., and Hatzimanikatis, V. (2007). Thermodynamics-based metabolic flux analysis. *Biophysical Journal*, 92, 1792–1805.

Lin, H., Bennett, G.N., and San, K.Y. (2005). Chemostat culture characterization of *Escherichia coli* mutant strains metabolically engineered for aerobic succinate production: a study of the modified metabolic network based on metabolic profile, enzyme activity, and gene expression profile. *Metabolic Engineering*, 7, 337–352.

Lun, D.S., Rockwell, G., Guido, N.J., Baym, M., Kelner, J.A., Berger, B., Galagan, J.E., and Church, G.M. (2009). Large-scale identification of genetic design strategies using local search. *Molecular Systems Biology*, 5, 296.

Mahadevan, R., Burgard, A.P., Famili, I., Van Dien, S., and Schilling, C.H. (2005). Applications of metabolic modeling to drive bioprocess development for the production of value-added chemicals. *Biotechnology and Bioengineering*, 10, 408–417.

Mahadevan, R. and Schilling, C.H. (2003). The effects of alternate optimal solutions in constraint-based genome-scale metabolic models. *Metabolic Engineering*, 5, 264–276.

Pharkya, P. and Maranas, C.D. (2006). An optimization framework for identifying reaction activation/inhibition or elimination candidates for overproduction in microbial systems. *Metabolic Engineering*, 8, 1–13.

Yang, L., Mahadevan, R., and Cluett, W.R. (2008). A bilevel optimization algorithm to identify enzymatic capacity constraints in metabolic networks. *Computers and Chemical Engineering*, 32, 2072–2085.