Fast Thermodynamically Constrained Flux Variability Analysis

Arne C. Müller1,2,4, Alexander Bockmayr1,3

1 Department of Mathematics and Computer Science, Freie Universität Berlin, Arnimallee 6, 14195 Berlin, Germany
2 International Max Planck Research School for Computational Biology and Scientific Computing (IMPRS-CBSC), Max Planck Institute for Molecular Genetics, Ihnestr 63-73, D-14195 Berlin, Germany
3 DFG-Research Center Matheon, Berlin, Germany
4 Berlin Mathematical School (BMS), Berlin, Germany

ABSTRACT

Motivation: Flux variability analysis (FVA) is an important tool to further analyze the results obtained by flux balance analysis (FBA) on genome-scale metabolic networks. For many constraint-based models FVA identifies unboundedness of the optimal flux space. This reveals that optimal flux solutions with net flux through internal biochemical loops are feasible, which violates the second law of thermodynamics. Such unbounded fluxes may be eliminated by extending flux variability analysis with thermodynamic constraints.

Results: We present a new algorithm for efficient flux variability (and flux balance) analysis with thermodynamic constraints, suitable for analyzing genome-scale metabolic networks. We first show that flux balance analysis with thermodynamic constraints is NP-hard. Then we derive a theoretical tractability result, which can be applied to metabolic networks in practice. We use this result to develop a new constraint programming algorithm Fast-tFVA for fast flux variability analysis with thermodynamic constraints (tFVA). Computational comparisons with previous methods demonstrate the efficiency of the new method. For FVVA, a speed-up of factor 30-300 is achieved. In an analysis of genome-scale metabolic networks in the BioModels database, we found that in 485 out of 716 networks additional irreversible or fixed reactions could be detected.

Availability and Implementation: Fast-tFVA is written in C++ and published under GPL. It uses the open source software SCIP and libSBML. There also exists a Matlab interface for easy integration into Matlab. Fast-tFVA is available from page.mi.fu-berlin.de/arnem/fast-tfva.html.

Contact: arne.mueller@fu-berlin.de, Alexander.Bockmayr@fu-berlin.de

Supplementary information: The proofs that are not contained in this article can be found in the supplementary material, which is available on the journal’s web site.

1 INTRODUCTION

Flux Balance Analysis (FBA) is a widely used method to analyze the capabilities of a metabolic network (Varma and Palsson, 1994; Mahadevan and Schilling, 2003; Price et al., 2004; Terzer et al., 2009; Durot et al., 2009; Schuster et al., 2007; Teusink et al., 2009; Orth et al., 2010). The strength of FBA is that it allows predicting growth rates of cells very accurately (Edwards et al., 2001). Flux Balance Analysis is based on the steady-state assumption, i.e., every metabolite that is produced, must also be consumed at the same rate. Flux through the network is enabled by exchange reactions, such as uptake of nutrients and production of biomass. Usually, next to bounds on the fluxes, stoichiometric constraints are the only constraints given, and optimal production of biomass can be computed by solving a linear program (LP).

Frequently, there is not only one optimal flux distribution that achieves optimal biomass production. FBA, however, computes only one such solution. Elementary flux modes (Schuster and Hilgetag, 1994; Schuster et al., 2000) or extreme pathways (Schilling et al., 2000) are commonly used tools to analyze the whole flux space. But, in practice, the number of elementary flux modes grows exponentially with the number of reactions in the network. Hence enumeration of elementary flux modes becomes infeasible for genome-scale networks. In addition, we are usually not interested in the raw data of all elementary flux modes (resp. extreme pathways), but only in specific properties (Schwartz and Kanehisa, 2006; Haus et al., 2008; Driouch et al., 2012; Orman et al., 2012). Therefore, sampling methods (Schellenberger and Palsson, 2009), elementary flux patterns (Kaleta et al., 2009) and flux variability analysis (Mahadevan and Schilling, 2003; Burgard et al., 2001) have been developed.

Flux Variability Analysis (FVA) determines the maximum and minimum values of all the fluxes that will satisfy the constraints and allow for the same optimal objective value. For example, it is known that FBA is very unreliable in predicting the flux value of by-products (Khannapho et al., 2008). In such cases, FVA can be applied to predict the range of possible by-product production rates under maximal biomass production, which can be linked to gene expression data (Bilu et al., 2006). Variations of FVA can also be used to determine blocked or unessential reactions (Burgard et al., 2004).
In FVA, however, the following problem arises: It can happen that the network contains internal cycles, i.e., there exist non-zero steady-state fluxes involving only internal reactions. In most metabolic models, only bounds on the flux value of exchange reactions are given (in addition to the zero bounds for the irreversible reactions). This leads to unbounded fluxes through reactions contained in such internal cycles, which is of course not realistic. To remove this issue, a general form of thermodynamic constraints can be added to the model (Beard et al., 2002; Qian and Beard, 2005), extending the standard non-negativity constraints for thermodynamic irreversibility of individual reactions. It has been shown that the flux values computed with thermodynamic constraints are more consistent, for example, with respect to gene expression data (Schellenberger et al., 2011a; Feist et al., 2007; Fleming et al., 2009; Price et al., 2006; Jol et al., 2012). In recent years there has been increasing interest in thermodynamic constraints, since they can also be used to link metabolite concentrations to flux modes (Kümml et al., 2006a,b; Hoppe et al., 2007; Henry et al., 2006; Singh et al., 2011).

In this paper, we present a new method for efficient thermodynamically constrained flux balance and flux variability analysis. We will work with the relaxed form of thermodynamic constraints, as it was introduced by Beard et al. (2004). There, thermodynamic constraints are formulated as follows:

\[
SJ = 0 \quad \text{steady-state assumption (1)}
\]

\[
\ell \leq J \leq u \quad \text{flux bounds, e.g. irrev. (2)}
\]

\[
\Delta \mu_i J_i < 0 \quad \forall i \in I \quad \text{thermodynamic constraint (3)}
\]

\[
\Delta \mu^T I = \mu^T S \quad \text{potential differences (4)}
\]

\[
J \in \mathbb{R}^\mathcal{R}, \quad \mu \in \mathbb{R}^\mathcal{M}, \quad \Delta \mu \in \mathbb{R}^\mathcal{I}.
\]

Here \( S \) denotes the stoichiometric matrix, \( J \) the flux vector, \( \mathcal{M} \) the set of metabolites, \( \mathcal{R} \) the set of all reactions, \( \mathcal{I} \) the set of exchange reactions, \( I = \mathcal{R} \setminus \mathcal{E} \) the set of internal reactions, and \( \mathcal{E} \) the set of exchange reactions. For the chemical potential of each metabolite (Gibbs free energy of formation) the operator \( ^T \) denotes transposition. If the concentrations and equilibrium constants are known (Mavrovouniotis, 1990; Jankowski et al., 2008; Noor et al., 2012a), it is possible to further constrain \( \mu \). However, since this information is often not available, we will work here without additional constraints on \( \mu \).

In a thermodynamically constrained flux balance problem, we maximize a linear objective function \( c^T J \), where \( c \in \mathbb{R}^\mathcal{R} \), with respect to the constraints (1)–(4). In thermodynamically constrained flux variability analysis (tFVA), we analyse the flux variability within the feasible (respectively optimal) solutions of a thermodynamically constrained flux balance problem, i.e., for all \( i \in \mathcal{R} \), we solve the optimization problem \( \max \{ \pm J_i \mid (1)-(4) \} \) (respectively \( \max \{ \pm J_i \mid (1)-(4), c^T J = \text{opt} \} \)).

To simplify notation we do not only use reactions as indices (e.g. \( J_i \) to denote flux through reaction \( i \)), but also sets of reactions to index sub-vectors. For example, \( J_S \) denotes the flux vector on the internal reactions only and \( S \) contains only the columns corresponding to internal reactions.

**Definition 1** (Thermodynamically Feasible Flux). A flux vector \( J \) is called thermodynamically feasible if there exists a vector \( \mu \) such that (1), (3), and (4) are satisfied. If additionally (2) holds, \( J \) will be called a thermodynamically feasible flux that satisfies bounds \( \ell \) and \( u \).

By multiplying \( \mu^T \) from the left side with \( S \), the potential differences for internal reactions are obtained. This is equivalent to the often found formulation \( K \Delta \mu = 0 \), where \( K \) is the null-space matrix of \( S \). The motivation behind Eq. 3 is that normally a chemical reaction carries flux if and only if it reduces Gibbs free energy (Beard et al., 2002; Alberty, 2003; Qian and Beard, 2005). Since many reactions are catalyzed by enzymes, however, it can happen that an enzyme is not present, for example because of regulatory control; and hence, the corresponding reaction is not possible and does not carry flux, even if there is a negative potential difference. Thus, zero flux in Eq. 3 is always allowed as well. Note that other formulations of thermodynamic constraints, for example the one used by Fleming et al. (2012), do not have this property. Hence, the different results have to be applied with care.

Using Boolean variables \( a \in \{0,1\}^\mathcal{I} \), a sufficiently large constant \( M > 0 \), and a small constant \( \varepsilon > 0 \), this formulation can be translated into a mixed integer linear program (MILP), where \( a_i = 0 \) indicates that a forward flux \( J_i > 0 \) on reaction \( i \) is not allowed, and \( a_i = 1 \) forbids a backward flux \( J_i < 0 \). This is done by replacing (3) with the inequalities (5) and (6):

\[
SJ = 0
\]

\[
\ell_i \leq J_i \leq u_i \quad \forall i \in \mathcal{R}
\]

\[
\ell_i (1-a_i) \leq J_i \leq u_i a_i \quad \forall i \in \mathcal{I}
\]

\[
-Ma_i + \varepsilon \leq \Delta \mu_i \leq M (1-a_i) - \varepsilon \quad \forall i \in \mathcal{I}
\]

\[
\Delta \mu^T = \mu^T S
\]

\[
J \in \mathbb{R}^\mathcal{R}, \mu \in \mathbb{R}^\mathcal{M}, \Delta \mu \in \mathbb{R}^\mathcal{I}, \quad a \in \{0,1\}^\mathcal{I}.
\]

This formulation is used in a similar form by Schellenberger et al. (2011a) for the COBRA toolbox and many others (Beard et al., 2004; Cogne et al., 2011; Hoppe et al., 2007; Henry et al., 2007). It can be shown that if \( S \) does not contain any zero-columns, this MILP formulation is equivalent to the original one (Müller, 2012). Although solving MILPs is NP-hard in general, practical tests have shown that current MILP solvers are able to optimize flux with respect to the above formulation also on genome-scale networks like the iAF1260 reconstruction of E. coli (Schellenberger et al., 2011a).

Beard et al. (2004) observed that a steady-state flux vector \( J \) (i.e., which satisfies Eq. 1) is thermodynamically feasible if and only if there is no internal cycle contained by \( J \). A sketch of the proof using oriented matroids can be found in (Beard et al., 2004). A detailed version using LP-duality is available in Müller (2012) and was also recently published by Noor et al. (2012b). This result is used by Yang et al. (2005) to detect reactions that are irreversible due to thermodynamic constraints. In particular, one of their methods simply runs tFVA.

This paper is organized as follows: In Sect. 2.1, we show that the thermodynamically constrained flux balance problem is NP-complete and thus, MILPs are an appropriate tool. In Sect. 2.2, we derive a tractability result and analyze how much this applies to current genome-scale models. These theoretical results are the backbone of a new algorithm, which is described in Sect. 3. In Sect. 4 we apply our new method to study the importance of thermodynamic constraints in the analysis of genome-scale metabolic networks.
2 METHODS

2.1 NP-completeness

In FVA, a series of optimization problems is solved. The flux through each reaction in the network is maximized and minimized. In the following, we will see that already determining whether there exists a positive flux through a given reaction is NP-complete. Thus, the optimization problem is NP-hard, and it is unlikely that an efficient algorithm exists for the general case.

Problem 1 (ThermoFLUX).

- Instance:
  - Metabolic network \( \mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \cup \mathcal{E}, S \in \mathbb{Q}^{M \times R}) \)
  - Objective reaction \( r \in R \)

- Question: Does a thermodynamically feasible flux \( J \geq 0 \) with \( J_r > 0 \) exist?

Theorem 1. ThermoFLUX is NP-complete.

The proof of this result can be found in the supplementary material.

2.2 Tractability

It was observed early on by Price et al. (2002) that a thermodynamically infeasible flux can be turned feasible by removing internal cycles. For the following tractability result it is important to understand when these internal cycles can be removed without changing the flux through the objective reactions. Reactions that are contained in internal cycles and reactions that are flux-forcing are problematic.

Definition 2. Let \( \mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \cup \mathcal{E}, S \in \mathbb{Q}^{M \times R}) \) be a metabolic network. A reaction \( r \in R \) is contained in an internal cycle if there exists a \( J \in \mathbb{R}^R \) with \( S_J = 0 \) and \( J_r > 0 \).

Given lower and upper flux bounds \( \ell, u \in \mathbb{R}^R \), a reaction \( r \) is called flux-forcing if \( \ell_r > 0 \) or \( u_r < 0 \).

For a linear objective function \( c \in \mathbb{R}^R \), a reaction \( r \) is called objective if \( c_r \neq 0 \).

Theorem 2. Let \( \mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \cup \mathcal{E}, S \in \mathbb{Q}^{M \times R}) \) be a metabolic network with lower and upper flux bounds \( \ell, u \in \mathbb{R}^R \) and a linear objective function \( c \in \mathbb{R}^R \). Let \( C \subseteq I \) be the set of reactions contained in internal cycles, let \( F \subseteq R \) be the set of reactions that are flux-forcing, and let \( O \subseteq R \) be the set of objective reactions.

Given a steady-state flux \( J \) with \( \ell \leq J \leq u \), a thermodynamically feasible flux \( J^* \) with \( c^T J^* = c^T J \) and \( \ell \leq J^* \leq u \) can be computed in polynomial time w.r.t. the size of the network \( N \) if \( C \cap (F \cup O) = \emptyset \).

The key to the proof of Thm. 2 is Alg. 1 which gives the wanted polynomial-time algorithm. The complete proof can be found in the supplementary material.

It should be noticed that this theorem is similar to the result by Fleming et al. (2012). While these authors use a different definition of thermodynamic feasibility, they also propose a method that keeps the flux through exchange reactions invariant.

Since we can compute optimal steady-state fluxes in polynomial time by solving an LP, it follows that we can compute an optimal thermodynamically feasible flux in polynomial time, if all of the reactions contained in internal cycles are neither flux-forcing nor contained in the objective function. In practice, however, there will be reactions contained in internal cycles and we have also to consider this case.

For the following result, we need to quantify how often a reaction is contained in internal cycles. For this, it comes in very handy that internal cycles are simply steady-state flux vectors that do not use exchange reactions. Hence, we can describe the space of all internal cycles by elementary flux modes (Schuster and Hilgetag, 1994; Schuster et al., 2000). The elementary flux modes that do not contain exchange reactions will be called internal circuits or simply circuits. They may also be interpreted in terms of oriented matroid theory, which was introduced into metabolic network analysis by Oliveira et al. (2001) and used by Beard et al. (2004) to describe the internal circuits. The number of internal circuits containing the given reaction will be the quantification measure.

We observed that the number of internal circuits is small for many genome-scale networks. See the supplementary materials for more details.

Theorem 3. If the number of internal circuits containing flux-forcing or objective reactions is bounded by a constant, the thermodynamically constrained flux balance (and flux variability) problem can be solved in polynomial time w.r.t. the size of the network.

Algorithm 1. This algorithm computes a thermodynamically feasible flux out of a possibly thermodynamically infeasible flux, if the conditions of Thm. 2 are satisfied. It runs in polynomial time. (\( \{ \cdot \} \) denotes a vector where all entries are 1.)

Input: A steady-state flux \( J \)
repeat
  \[ I^+ := \{ i \in I : J_i \geq 0 \}, \]
  \[ I^- := \{ i \in I : J_i \leq 0 \}. \]
  \[ L := \arg\max \left\{ \frac{1}{2} L_{1+} - \frac{1}{2} L_{1-} : S_L = 0, \right. \]
  \[ J_{1-} \leq L_{1-} \leq 0, \]
  \[ J_{1+} \geq L_{1+} \geq 0 \}
until \[ \frac{1}{2} L_{1+} - \frac{1}{2} L_{1-} = 0 \]
return \( J \)

Fig. 1. Blocking set (dashed arrows) for a given thermodynamically feasible flux (marked in black).

Fig. 2. If a “wrong” blocking set is chosen, only suboptimal flux (here: no flux) may be possible.
Algorithm 2. General solving procedure

Input: objective function $c$, lower bounds $lb$, upper bounds $ub$

maxFlux := $-\infty$

for all blocking set $B$ do

$\ell := lb$, $u := ub$

for each $(i, +) \in B$ do $u_i := 0$

for each $(i, -) \in B$ do $\ell_i := 0$

$J = \text{solve ordinary FBA with bounds } \ell, u$

maxFlux = max(maxFlux, $c^T J$)

end for

return maxFlux

In case of the E. coli iAF1260 model, the bound of Thm. 3 tells us that we only need to analyze at most 1680 different blocking sets (for optimization on one reaction without internal flux-forcing reactions).

3 IMPLEMENTATION

To implement Alg. 2 and use the result of Thm. 3, we still have to find a way for enumerating all blocking sets. To do this, we used the constraint integer programming (CP) framework Scip (Achterberg, 2009). Scip can not only solve mixed integer programs (Wolsey, 1998), but also offers an interface for adding more general and complex constraints. Hence, we implemented a constraint handler and primal heuristic for Scip.

Our algorithm (see Alg. 3) works directly on the flux variables and does not use artificial Boolean variables. It basically implements a branch-and-bound strategy. As we have seen in Thm. 3, we need not block all internal circuits, but only those that contain objective or flux-forcing reactions.

- With highest priority, the constraint handler branches on circuits that contain objective reactions. If this is achieved, every flux through objective reactions must be sourced by exchange reactions. This usually bounds the flux to realistic values and we may be able to reject the current blocking set, because a better solution has already been found.
- If no circuits containing objective reactions are left, we branch on circuits containing flux-forcing reactions.
- It never happens that we need to branch on circuits that contain neither objective reactions nor flux-forcing reactions. This is because of the heuristic we also implemented.

The heuristic basically runs Alg. 1. This means, if the network contains no circuits with objective or flux-forcing reactions, the heuristic will output an optimal flux and the current branch of the search tree does not need to be analyzed further. If the network contains circuits with objective reactions, we usually have a very large flux through an internal cycle and thus, a very bad dual bound. Although Alg. 1 may also find an optimal solution in this case, the solver will not know it (the dual bound will be larger than the value of the solution found). To prove optimality, the solver will have to do the branching nonetheless. This is why we do not run the heuristic in this case. The heuristic may also fail if the network does contain circuits with flux-forcing reactions and no circuits with objective reactions. However, we may be lucky and Alg. 1 produces a solution that still satisfies the flux bounds. This solution is thus proven to be optimal. Therefore, we implemented a slight modification of Alg. 3 that additionally runs the heuristic if there exist circuits with flux-forcing reactions, but no circuits with objective reactions.

Thus, the heuristic is run at those nodes of the search-tree that do not contain any circuits with objective reactions. This way, we usually only need to branch on circuits containing objective reactions, sometimes on circuits containing flux-forcing reactions and never on circuits containing none of the two.

Next to the actual solving routine, the result of Thm. 2 can also be used to speed-up iFVA significantly. Gudmundsson and Thiele (2010) observed that for ordinary FVA, a speed-up of factor 100 could be achieved by simply warm starting the LP computation necessary for FBA. Previous implementations always created new instances of the LP problem that the LP-solver had to solve from scratch. In warm starting, the LP-solution of the previous iteration is used as the starting point of the simplex algorithm used by the LP-solver. In the case of iFVA, however, we do not solve LPs anymore, hence this result is not directly applicable. On the other hand, many genome-scale metabolic models only contain few reactions involved in internal cycles, see Fig. 3 and Tab. 1. By Thm. 2 it follows that for most reactions the LP-solution can easily be transformed into a thermodynamically feasible one without having to start any MILP- or CP-solver. Thus, the warm-starting effect can also be used for iFVA.

4 DISCUSSION

4.1 Run time

We tested the correctness and run time of our implementation on various networks of the BiGG-database (Schellenberger et al., 2011b). For iFVA, we compared our results with those generated by the COBRA toolbox and obtained a difference of the order
10^{-3}. This difference is likely due to numerical issues of the big-M formulation (Bradley et al., 1977), since the results of our code for different orderings of the reactions and metabolites only vary in the order of 10^{-8}. The E. coli iAF1260 model was used for a more detailed analysis. We compared the CP approach to two different kinds of MILP formulations found in the literature; we did not compare it to nonlinear formulations, since most nonlinear solvers do not return globally optimal solutions. All computations were run on an Intel Core i5-2400S (2.5 GHz, 4 cores, 6 MiB L3 cache) with 4GB RAM. Both MILP formulations were solved using Gurobi (www.gurobi.com), the CP formulation was solved using SCIP together with the public-domain solver S O P L E X (Achterberg, 2009). As can be seen in Fig. 3 of the supplementary material, the CP-approach outperforms the MILP formulations in nearly all cases.

We also performed run time tests for tFVA and compared our algorithm to the one implemented in the COBRA-Toolbox (Schellenberger et al., 2011a). The CP algorithm is 30-300 times faster than the COBRA-Implementation (see Tab. 1). The only trouble maker is the H. sapiens reconstruction that did not finish within 2 hours in our implementation. We were also not able to confirm the run time result by Schellenberger et al. (2011a) on that network, because COBRA (version 2.0.3) with Gurobi 5.0.0 terminated with an error message, probably due to numerical instabilities of the MILP formulation in the Gurobi solver.

4.2 Irreversible and fixed flux rates due to thermodynamics

In a second study, we analyzed how much information on irreversibility and fixed flux rates of reactions can be gained by adding thermodynamic constraints. Fixed reactions are sometimes the focus of network analysis, as in Hädicke et al. (2011). We used the genome-scale networks in the BioModels database (http://www.ebi.ac.uk/biomodels-main, available on 17 Sep 2012). Nearly all of these networks contain reactions in internal cycles (see Fig. 3). Hence, nearly every network of the BioModels database has reactions where FVA cannot predict bounds different from those given in the original FBA problem. These bounds tend to be very uninformative, since usually the only small bounds are given on exchange reactions. tFVA, however, will compute bounds for these internal reactions that depend on the bounds on the exchange reactions, in particular the bounds on nutrient uptake. Therefore, the bounds obtained by tFVA for reactions contained in internal cycles will be better than the bounds computed by ordinary FVA by orders of magnitude. Irreversible reactions and reactions with fixed fluxes are a special kind of reactions with improved bounds, which we now investigate in more detail.

As explained in Sect. 2, our algorithm works best if the number of reactions in internal cycles is small. In addition, as it can be seen in Fig. 3, the genome-scale networks in the BioModels database may be divided into two categories: networks with less than 600 reactions in internal cycles and networks with significantly more reactions. In what follows, we analyzed those networks with less than 600 reactions in internal cycles. For each network we ran FVA and tFVA on the whole flux space. If the model also contained an objective function on some reaction (e.g. biomass production), we also analyzed the optimal flux space with FVA and tFVA. We ran the computations of FVA and tFVA for at most 30 minutes.

We obtained that 829 (45% of all genome-scale networks in the BioModels database) had less than or equal to 600 reactions in internal cycles. Of these we were able to analyze 716 networks in the given time limit. For the whole (resp. the optimal) flux space, we obtained the following results:

- in 386 (resp. 387) networks, flux through at least 1 reaction was additionally fixed due to thermodynamic constraints.
- In 481 (resp. 485) networks, at least 1 additional reaction was detected to be irreversible due to thermodynamic constraints.

In Fig. 4 we provide further information on the number of additionally fixed or irreversible reactions when analyzing the whole flux space. The plot looks similar for the optimal flux space. Additional details can be found in the supplementary material.

Finally, we ran the same analysis for the networks of the BiGG-database and obtained that in all of them, except S. aureus iSB619, additional fixed and irreversible reactions were detected, see Table 2. In the optimal flux space, also in S. aureus iSB619 additional fixed and irreversible reaction were found.

These results show that thermodynamic constraints provide useful information even for well-curated models like those in the BiGG-database. Using our algorithm, tFVA can now be performed in a routine manner.

4.3 Conclusion

We presented a new algorithm for thermodynamically constrained flux variability analysis. Compared to previous implementations, an enormous speed-up was obtained. tFVA can now be performed for a large number of networks and as a subroutine for other methods. As shown in our experiments, many network reconstructions, in particular also well-curated models, may be improved by executing our algorithm.
Table 1. Running Times of FVA on different metabolic networks. FVA was performed on the whole flux space and on the subspace of the fluxes with optimal biomass production rate. The COBRA implementation used the Gurobi solver. The COBRA method sometimes failed to produce a solution, because Gurobi returned infeasible for problems that are actually feasible.

<table>
<thead>
<tr>
<th>Model</th>
<th>method</th>
<th>internal reactions</th>
<th>whole flux space</th>
<th>only optimal fluxes w.r.t. biomass production</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>(in internal circuits)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E. coli iJR904</td>
<td>CP</td>
<td>19</td>
<td>1075 (40)</td>
<td>11s</td>
</tr>
<tr>
<td></td>
<td>COBRA</td>
<td></td>
<td></td>
<td>963s</td>
</tr>
<tr>
<td>E. coli iAF1260</td>
<td>CP</td>
<td>38</td>
<td>2382 (68)</td>
<td>71s</td>
</tr>
<tr>
<td></td>
<td>COBRA</td>
<td></td>
<td></td>
<td>11162s</td>
</tr>
<tr>
<td>E. coli iJO1366</td>
<td>CP</td>
<td>46</td>
<td>2583 (76)</td>
<td>92s</td>
</tr>
<tr>
<td></td>
<td>COBRA</td>
<td></td>
<td></td>
<td>25869s</td>
</tr>
<tr>
<td>H. pylori iIT341</td>
<td>CP</td>
<td>8</td>
<td>554 (22)</td>
<td>4s</td>
</tr>
<tr>
<td></td>
<td>COBRA</td>
<td></td>
<td></td>
<td>122s</td>
</tr>
<tr>
<td>H. sapiens Recon. 1</td>
<td>CP</td>
<td>?</td>
<td>3742 (958)</td>
<td>&gt; 2h</td>
</tr>
<tr>
<td></td>
<td>COBRA</td>
<td></td>
<td></td>
<td>Error</td>
</tr>
<tr>
<td>M. Barker iAF692</td>
<td>CP</td>
<td>31</td>
<td>690 (30)</td>
<td>7s</td>
</tr>
<tr>
<td></td>
<td>COBRA</td>
<td></td>
<td></td>
<td>204s</td>
</tr>
<tr>
<td>M. tuberculosis iNJ661</td>
<td>CP</td>
<td>66</td>
<td>1025 (53)</td>
<td>22s</td>
</tr>
<tr>
<td></td>
<td>COBRA</td>
<td></td>
<td></td>
<td>844s</td>
</tr>
<tr>
<td>S. aureus iSB619</td>
<td>CP</td>
<td>3</td>
<td>743 (7)</td>
<td>3s</td>
</tr>
<tr>
<td></td>
<td>COBRA</td>
<td></td>
<td></td>
<td>179s</td>
</tr>
<tr>
<td>S. cerevisiae iND750</td>
<td>CP</td>
<td>43</td>
<td>1266 (64)</td>
<td>44s</td>
</tr>
<tr>
<td></td>
<td>COBRA</td>
<td></td>
<td></td>
<td>1662s</td>
</tr>
</tbody>
</table>

Our current algorithm works without additional information on metabolite concentrations. However, in more and more cases, such information becomes available. In future work, we plan to extend our method by incorporating metabolite concentrations and other experimental data (Kümml et al., 2006a; Shinar and Feinberg, 2010; Pey et al., 2011).

**ACKNOWLEDGEMENT**

This work was funded by the Berlin Mathematical School in terms of a PhD stipend.

**REFERENCES**


noor, e., bar even, a., flamholz, a., lubling, y., davidi, d., and milo, r. (2012a). an integrated open framework for thermodynamics of reactions that combines accuracy and coverage. bioinformatics, 28(15), 2037-2044.


orman, m. a., androulakis, i. p., berthiaume, f., and irerapitou, m. g. (2012). metabolic network analysis of perfused livers under fed and fasted states: Incorporating thermodynamic and futile-cycle-associated regulatory constraints. journal of theoretical biology, 293, 101-110.


