Finding Minimal Reaction Sets in Large Metabolic Pathways

Takehide Soh[1,2] and Katsumi Inoue[3,2]

[1] Research Fellow of the Japan Society for the Promotion of Science
[2] Department of Informatics, The Graduate University for Advanced Studies, Chiyoda-ku, Tokyo, Japan
[3] Principles of Informatics Division, National Institute of Informatics, Chiyoda-ku, Tokyo, Japan

Abstract

In systems biology, identifying vital functions like glycolysis from a given metabolic pathway is important to understand living organisms. In this paper, we particularly focus on the problem of finding minimal sub-pathways producing target metabolites from source metabolites. We represent laws of biochemical reactions in propositional formulas and use a minimal model generator based on a state-of-the-art SAT solver. An advantage of our method is that it can treat reversible reactions represented in cycles. Moreover recent advances of SAT technologies enables us to obtain solutions for large pathways. We have applied our method to a whole *Escherichia coli* metabolic pathway. As a result, we found 5 sets of reactions including the conventional glycolysis sub-pathway described in a biological database EcoCyc.

1 Introduction

Living organisms are kept alive by a huge number of chemical reactions. In *systems biology*, interactions of such chemical reactions are represented in a network called *pathway*. Analyses of pathways have been active research field in the last decade and several methods have been proposed [7, 17]. A longstanding approach is to represent pathways as systems of differential equations. This method allows detailed analyses e.g. concentrations of each metabolite with time variation. However, it is not applicable to a large network due to its difficult parameter tuning. This is a problem because scalability is an important feature for macroscopical analyses of complex networks like cells, organisms and life, which is a fundamental goal in systems biology. Therefore other methods aiming for scalable and abstracted analyses have been proposed [2, 3, 12, 11, 15]. Although these methods are different from each others in these problem formalization and solving methods, their purpose is the same, that is, to identify biologically meaningful sets of reactions from a given pathway.

One of these methods proposed by Schuster *et al.* is called elementary mode analyses. It focuses on a flux distribution, which is computed by matrix calculus, corresponding to a set of reactions in metabolic pathways [15]. This method can treat multi-molecular reactions while taking into account stoichiometry, and its computational scalability is enough to analyze large pathways. However it tends to generate a large number of solutions without ordering e.g. over 20000 solutions for a pathway including 100 reactions. Even though found solutions are potentially interesting, analyzing all of them through biological experiments would be infeasible task. We thus need a method which generates lower number of solutions keeping its quality. Another approach relying on graphs is proposed by Croes *et al.* [3]. They formalized their problem using a weighted graph and ran a depth-first search algorithm to find the lightest paths from a source compound to a target compound. Planes and Beasley proposed to solve the same problem using a constraint-based method [11]. An advantage of these two methods is that an evaluation of the quality of the solution is provided. We can then choose an objective value to reduce the number of solutions that should be provided to biologists. However, this approach can only generate paths while sub-pathways would be a more natural representation. The approach thus sometimes generates not valid paths from a biological viewpoint because it can easily take non-meaningful shortcuts via common metabolites, such as water, hydrogen and adenosine triphosphate (ATP).

 A. Dovier, A. Dal Palù, S. Will (eds.), WCB10 (EPiC Series, vol. 4), pp. 38–52

In this paper, we propose a new analysis method for metabolic pathways which finds sub-pathways producing a set of target metabolites from a set of source metabolites. In particular, we focus on finding minimal sub-pathways which has the property of not containing any other feasible sub-pathways, that is, intuitively, all elements in each minimal sub-pathway are qualitatively essential to produce target metabolites. We represent laws of biochemical multi-molecular reactions in propositional formulas and translate the problem into conjunctive normal form (CNF) formulas. We then use a model generator based on state-of-the-art SAT solver to solve the problem efficiently. Recent progresses in SAT domain now make it possible to apply our method to huge pathways. In realistic metabolic pathways, there are a lot of reversible reactions. Previous approaches thus needed pre-processing or post-processing, which is possibly costly, to deal with reversible reactions in a pathway [11, 16]. We also show how our method treats such reversible reactions by a minimal model generation. For evaluation, we compared our method with previously proposed approaches [1, 11] for a simplified pathways of *E. coli*, which consists of 880 reactions. As a result, our method identified 10 experimentally elucidated sub-pathways, while the previous methods identified at most 8 sub-pathways. We also tested our method with a whole *Escherichia coli* (*E. coli*) pathway consisting of 1777 reactions.

In the reminder of this paper, we explain propositional formulas and its minimal models in Section 2. In Section 3, we explain the sub-pathway finding problem and the difference from the previously proposed path finding problem. We show the translation from the sub-pathway finding problem into propositional formulas in Section 4. In Section 5, we show the experimental result. In Section 6 and 7 respectively discuss related work and future work.

2 Propositional Formulas and Minimal Model Generation

This section reviews propositional formulas and its *minimal models*. Let $V = \{v_1, v_2, \ldots, v_i\}$ be a set of propositional *variables*. A *literal* is a propositional variable v_i or its negation $\neg v_i$. A *clause* is a disjunction of literals. A *conjunctive normal form (CNF) formula* is a conjunction of clauses and is also identified with a set of clauses. The truth value of a propositional variable is either $true$ (T) or $false$ (F). A *(partial) truth assignment* for V is a function $f : V \to \{T, F\}$. A literal v_i is said to be *satisfied* by a truth assignment f if its variable is mapped to T; a literal $\neg v_i$ is satisfied by a truth assignment f if its variable is mapped to F. A clause is satisfied if at least one of its literals is satisfied. A *model* for a CNF formula Ψ is a truth assignment f where all clauses are satisfied. Models can also be represented in the set of propositional variables to which it assigns true. For instance, the model assigning v_1 to $true$, v_2 to $false$, v_3 to $true$ is presented by the set $\{v_1, v_3\}$. In this manner, we can compare two models by set inclusion. We here give the following two definitions [8]:

Definition 1. Let V_p, V_1 and V_2 be sets of propositional variables. Then, V_1 is said to be *smaller* than V_2 with respect to V_p if $V_1 \cap V_p \subset V_2 \cap V_p$ holds.

Definition 2. Let Ψ be a propositional formula, V_p a set of propositional variables, and I a model of Ψ. Then I is a *minimal model of Ψ with respect. to V_p* when there is no model smaller than I with respect to V_p.

Example. Suppose that Ψ is a propositional formula $(v_1 \vee v_2) \wedge (\neg v_1 \vee \neg v_2) \wedge (\neg v_2 \vee v_3)$. Then all models of Ψ are $\{v_1\}$, $\{v_2, v_3\}$, $\{v_1, v_3\}$ and the minimal models are $\{v_1\}$ and $\{v_2, v_3\}$. Please note that minimality does not depends on the number of the elements, however depends on the inclusion relation between models, in other words, we focus on subset minimal models rather than numerical minimal models. Niemelä reported a method to find the minimal models for propositional logic with ECLiPSe Prolog [10]. Koshimura *et al.* also reported a minimal model generator based on SAT solvers [8] which we use in this paper. They provided the following theorem [10, 8]:

Minimal Model Generation Procedure (Ψ, V_p)
begin
  $\Sigma := \emptyset$;
  loop
    (res, I) = **Solve**(Ψ) ;
    if res = `UNSAT` **then** return Σ ;
    else
      $V_x := I \cap V_p$;
      $V_y := \overline{I} \cap V_p$;
      $\Psi_c := \Psi \wedge \left(\bigvee_{x_i \in V_x} \neg x_i\right) \wedge \left(\bigwedge_{y_j \in V_y} \neg y_j\right)$;
      (res, V_c) = **Solve**(Ψ_c) ;
      if res = `UNSAT` **then** $\Sigma := \Sigma \cup \{I\}$;
      $\Psi := \Psi \wedge \left(\bigvee_{x_i \in V_x} \neg x_i\right)$;
end

Figure 1: Procedure of Generating Minimal Models

Theorem 1. Let Ψ be a CNF formula, I be a model of Ψ, and V_p be a set of propositional variables. I is a minimal model of ψ with respect to V_p iff a formula $\Psi_c = \Psi \wedge \neg(x_1 \wedge x_2 \wedge \ldots \wedge x_i) \wedge \neg y_1 \wedge \neg y_2 \wedge \ldots \wedge \neg y_j$ is unsatisfiable, where $I \cap V_p = \{x_1, x_2, \ldots, x_i\}$, $\overline{I} \cap V_p = \{y_1, y_2, \ldots, y_j\}$.

We review a procedure of a minimal model generation [8] in Figure 1. The inputs of the procedure are a propositional formula Ψ and a set of propositional variables V_p. The output is a set Σ of minimal models. The function `Solve` corresponds to SAT solvers which return `SAT` and its model when a given formula is satisfiable. The function returns `UNSAT` otherwise.

3 Path Finding and Sub-pathway Finding

This section provides the detail of an existing problem *path finding problem* and formalises *sub-pathway finding problem* on which we are focusing. Let $M = \{m_1, m_2, \ldots, m_e\}$ be a set of metabolites, $R = \{r_1, r_2, \ldots, r_f\}$ a set of chemical reactions, and $A \subseteq (R \times M) \cup (M \times R)$ a set of arcs. A pathway is represented in a directed bipartite graph $G = (M, R, A)$ where M and R are two sets of nodes, A is a set of arcs. A metabolite $m \in M$ is called a *reactant* of a reaction $r \in R$ if there is an arc $(m, r) \in A$. On the other hand, a metabolite $m \in M$ is called a *product* of a reaction $r \in R$ if there is an arc $(r, m) \in A$. Let m_s be a source metabolite and m_t a target metabolite. Let $A_p = \{a_1, a_2, \ldots, a_n\}$ be a ordered set of arcs of a given pathway. In this paper, we call a simple and elementary path between m_s and m_t as a *path* which is represented in a ordered set A_p.

Example. Figure 2 shows a simple example of a directed bipartite graph representation of a pathway. Circle nodes and square nodes represent metabolites and reactions, respectively. For instance, `GLC-6-P` and `NADP` are reactants of the reaction `R1` and `PROTON` and `D-6-P-GLUCONO-DELTA-LACTONE` are products of the reaction `R1`. There are two paths between `RIBULOSE-5P` and `GAP`; one uses `R5` and `R6`, and the other uses `R4` and `R6`.

Path Finding Problem. The path finding problem which has been studied in the literature [3, 11, 12] is given as follows.

Definition 3. *Path Finding Problem*

Input Given by 6-tuple (M, R, A, w, m_s, m_t), where $M = \{m_1, m_2, ..., m_e\}$ is a set of metabolites, $R = \{r_1, r_2, ..., r_r\}$ is a set of reactions, $A \subseteq (R \times M) \cup (M \times R)$ is a set of arcs, $w : A \to Z^+$ is a mapping representing weights, $m_s \in M$ is a source compound and $m_t \in M$ is a target compound.

Output k-lightest paths such that $\sum_{a_i \in A_p} w(a_i) \leq k$

An important factor of the problem is a mapping w. The problem exactly corresponds to find k-shortest paths in the case of $w : A \to \{1\}$. However, solutions of the problem frequently include unexpected shortcuts [3]. To overcome this problem, Croes *et al.* proposed a mapping which is based on degree of the nodes of metabolites, that is, common compounds, such as water and hydrogen, are avoided because those have high degree. They had a comparison between three graphs. One is called raw graph which is original graph without any weights. Another one is filtered graph which omits the selected metabolites which have high degree from the raw graph. The other one is the weighted graph. Their method with the weighted graph successfully obtained better accuracy than former two graphs. An advantage of path finding approach is to be able to utilize existing graph-search algorithms. These algorithms are scalable enough to a whole pathway networks. However, there is still remaining problems of shortcuts. Figueiredo *et al.* summarised problems for path finding approach [3, 12] by a specific example [4]. For instance, graph-based approaches outputs two paths between `GLC-6-P` and `GAP` in the pathway. One is throughout `R1`, `R2`, `R3`, `R5` and `R6`. Another one uses `R4` and instead of `R5`. Although these paths give us one aspect of pathways, it is not easy to obtain reactions which are mainly occurred while target metabolites are produced. Considering a reaction `R6`, it needs `XYLULOSE-5-PHOSPHATE` and `RIBOSE-5P` as reactants. However, each obtained path cannot reflect this reaction law.

Sub-pathway Finding Problem. To overcome these problems, we propose and formalise a new problem called the *sub-pathway finding problem*. We here give additional terminology to define the problem.

Let $s : R \to 2^M$ be a mapping from a set of reactions to a set of metabolites such that $s(r) = \{m \in M | (m, r) \in A\}$ represents the set of metabolites which are needed for activating a reaction r. Let $p : R \to 2^M$ be a mapping from a set of reactions to a set of metabolites such that $p(r) = \{m \in M | (r, m) \in A\}$ represents the set of metabolites which are produced by a reaction r. Let s^{-1} and p^{-1} be inverse mappings of s and p, respectively. Let t be an integer variable representing a time and e be an integer value for a variable t. Let $M' \subseteq M$ be a subset of metabolites. A metabolite $m \in M$ is *producible* at a time $t = 0$ from M' if $m \in M'$ holds. A reaction $r \in R$ is *activatable* at a time $t = e\,(0 < e)$ from M' if $\forall m \in s(r)$ is producible at a time $t = e - 1$ from M'. A metabolite $m \in M$ is *producible* at a time $t = e\,(0 < e)$ from M' if $m \in p(r)$ holds for at least one reaction r which is activatable at a time $t = e$ from M'. If r is activatable at a time $t = e$ then r is activatable at a time $t = e + 1$. If m is producible at a time $t = e$ then m is producible at a time $t = e + 1$. Let $M_i \subseteq M$ be a subset of metabolites representing initial metabolites, $M_s \subseteq M$ a subset of metabolites representing source metabolites and $M_t \subset M$ a subset of metabolites representing target metabolites. Please note that we distinguish M_s from M_i. Every metabolite $m \in M_i$ represents universal metabolites which are always available in pathways, such that `WATER`, `ATP` and `PROTON`. On the other hand, M_s and M_t represent particular source metabolites and target metabolites in which we are interested, respectively.

Definition 4. Let π be a 6-tuple (M, R, A, M_i, M_s, M_t) and $G = \{M, R, A\}$ a graph. A sub-graph G' of G is a *sub-pathway* of π if $G' = (M', R', A')$ and it holds the following conditions (i), (ii) and (iii): (i) $M_s \subset M'$ and $M_t \subset M'$, (ii) For every $m \in M'$, m is producible from $M_i \cup M_s$ at a time $t \geq e$ for some $e \in Z^+$, (iii) For every $r \in R'$, r is activatable from $M_i \cup M_s$ at a time $t \geq e$ for some $e \in Z^+$. In

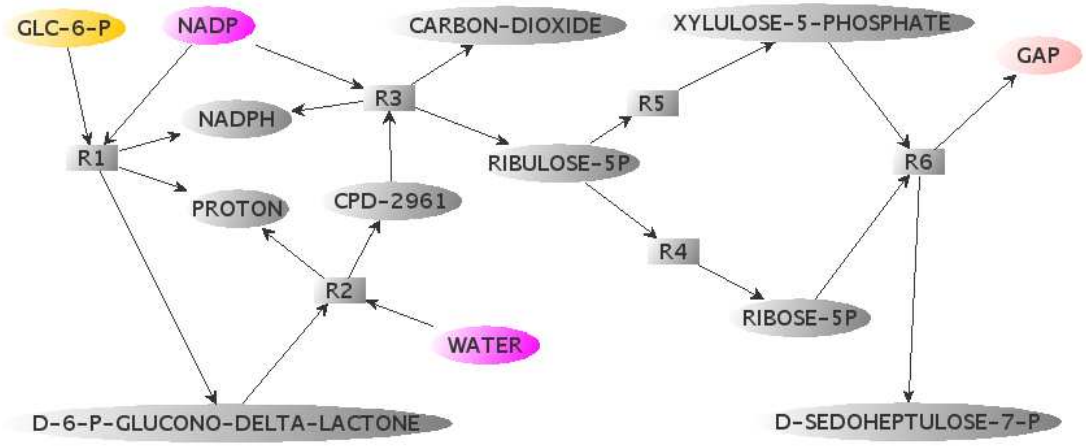

Figure 2: A Sub-pathway of the E. coli Pathway

addition, a sub-pathway G' is called *minimal* if it holds that (vi) there is no sub-pathway G'' such that $G'' \subset G'$.

Definition 5. *Sub-pathway Finding Problem*

Input Given by a 6-tuple $\pi = (M, R, A, M_i, M_s, M_t)$, where $M = \{m_1, m_2, ..., m_x\}$ is a set of metabolites, $R = \{r_1, r_2, ..., r_y\}$, $A \subseteq (R \times M) \cup (M \times R)$ is a set of arcs, $M_i \subset M$ is a set of initial compounds, $M_s \subset M$ is a set of source compounds, $M_t \subset M$ is a set of target compounds.

Output All minimal sub-pathways of π.

In practice, we compute more restricted solutions of the problem since the number of all minimal sub-pathways tends to be large. We describe how to restrict solutions in the next session. The solution of the example shown in Figure 2 is different between the path finding problem and sub-pathway finding problem. While the solution of the path finding problem is two paths, the solution of the sub-pathway finding problem is the one sub-pathway: `R1`, `R2`, `R3`, `R4`, `R5` and `R6`. Please note that the reaction `R6`, `XYLULOSE-5-PHOSPHATE` and `RIBOSE-5P` are needed as reactants to produce `GAP` and `D-SEDOHEPTULOSE`. The solution correctly reflect the law of the reaction `R6`. However path finding approach returns the activation of `R6` without producing both reactants. Obviously, the output of the sub-pathway finding problem reflects a biological law of reactions and is natural representation. More details of the problem the path finding is discussed in the literature [4].

4 Translation into Propositional Formulas

4.1 Translation

This section provides a translation for a 6-tuple π. Let e be integer for a time t. Let $rt_{n,e}$ be a propositional variable which is *true* if a reaction $r_n \in R$ is activatable at a time $t = e$ and later. Let $mt_{i,e}$ be a propositional variable which is *true* if a metabolite $m_i \in M$ is producible at a time $t = e$ and later. For each reaction and metabolite, we have the following supplemental formulas ψ_s:

$$rt_{n,e-1} \rightarrow rt_{n,e}, \quad mt_{i,e-1} \rightarrow mt_{i,e}.$$

For each reaction r_n, we have the following formula representing that if a reaction r_n is activatable at a time $t = e$ then its reactants must be producible at a time $t = e - 1$.

$$rt_{n,e} \rightarrow \bigwedge_{m_i \in s(r_n)} mt_{i,e-1} \tag{1}$$

For each reaction r_n, we have the following formula representing that if a reaction r_n is activatable at a time $t = e$ then its products must be producible at a time $t = e$.

$$rt_{n,e} \rightarrow \bigwedge_{m_j \in p(r_n)} mt_{j,e} \tag{2}$$

In a naive way, above two formulas are generated by every time t for every reaction. However it results in the expansion of translated clauses. We thus need to reduce the size of the translated formulas. Let $n()$ be a mapping from a set to the number of elements of the set. A time $t = e$ is called the *earliest activatable time* of a reaction $r \in R$ if r cannot be activatable at a time $0 < t < e$ and can be activatable $e \leq t$. Let $M' = M_s \cup M_i$ be a set of metabolites. Let d and n be integers. Let R' be the set of reactions which are activatable from M'. Let T be a set of integers $\{1, \ldots, n(R')\}$. Let $f_e : R' \rightarrow T$ be a mapping from a set of reactions to a set of integers representing each reaction $r_i \in R$ and its earliest activatable time $e \in T$. The mapping f_e is also represented in a set of pairs (r_i, e) of $r_i \in R$ and $e \in T$. We here show a procedure to form the mapping f_e in Figure 3. Let d_{max} be a constant represent the output integer value d of the procedure. Please note that this procedure can be done in polynomial time. This procedure can be seen a filtering method for a given π, that is, it omits the reactions which are not activatable from M'. Moreover, the earliest activatable time is useful to reduce the size of translated formulas. If e is the earliest activatable time for a reaction r obviously do not need to consider a time $t < e$ for the reaction. However the size of translated formulas still tends to be large.

Let $f_u : R' \rightarrow T$ be a bijection from a set of reactions to a set of integers representing each reaction and its *unique time*. In Figure 4, we show a procedure to construct the bijection f_u. To complete the procedure, we need to consider how to sort a set $\{r_i \mid (r_i, d) \in f_e\}$. We use a mapping $f_s(r_i) = \Sigma_{m_j \in s(r_i)} deg(m_j)$ where $deg(m_j)$ represents the outdegree of the node m_j. We sort a set $\{r_i \mid (r_i, d) \in f_e\}$ according to increasing order of $f_s(r_i)$. This sorting procedure place reactions, which consume low outdegree metabolites, earlier. We assumed that such a reaction is easier to be activatable because it may have a less number of competitive reactions which possibly dominates the compound.

For each reaction r_n and its unique time $f_u(r_n)$, we have the third formula representing that if a reaction r_n is not activatable then metabolites $m_j \in p(r_n)$ keeps its state from a time $f_u(r_n) - 1$.

$$\neg rt_{n,f_u(r_n)} \rightarrow \bigwedge_{m_j \in p(r_n)} \left(\neg mt_{j,f_u(r_n)-1} \rightarrow \neg mt_{j,f_u(r_n)}\right) \tag{3}$$

Please note that this formula does not mean that if r_n is not activatable then metabolites $m_j \in p(r_n)$ is not producible for any time. Some of those metabolites can be made to producible at a different time by some reactions since each reaction has its unique time. We have the following formula D_{r_n}, which is given as conjunction of the formulas (1), (2) and (3):

$$D_{r_n} = \left(rt_{n,f_u(r_n)} \rightarrow \bigwedge_{m_i \in s(r_n)} mt_{i,f_u(r_n)-1} \wedge \bigwedge_{m_j \in p(r_n)} mt_{j,f_u(r_n)}\right) \wedge \left(\neg rt_{n,f_u(r_n)} \rightarrow \bigwedge_{m_j \in p(r_n)} \left(\neg mt_{j,f_u(r_n)-1} \rightarrow \neg mt_{j,f_u(r_n)}\right)\right) \tag{4}$$

Activatable Time (M')
begin
  $d := 0$;
  while $(M' \neq \emptyset)$
    mark $\forall m_i \in M'$ as visited;
    $M'' = \emptyset$;
    $d := d + 1$;
    loop for $m_i \in M'$
      loop for $r_j \in s^{-1}(m_i)$
        if $r_j \notin \{r_k \mid (r_k, n) \in f_e, n \leq d\}$ and $\forall m_k \in s(r_j)$ is visited **then**
          $f_e := f_e \cup \{(r_j, d)\}$;
          loop for $m_k \in p(r_j)$
            if m_k is not visited **then**
              $M'' := M'' \cup \{m_k\}$;
    $M' := M''$;
return (f_e, d);
end

Figure 3: Procedure for f_e

Unique Time (f_e)
begin
  $u := 0$;
  loop for $d \in \{1, \ldots, d_{max}\}$
    R_{sorted} := **sort** $\{r_i \mid (r_i, d) \in f_e\}$;
    loop for $r_j \in R_{sorted}$
      $u := u + 1$;
      $f_u := f_u \cup \{(r_j, u)\}$;
return f_u;
end

Figure 4: Procedure for f_u

According to our translation, the number of the elements of $f_u(r_n)$ is the number of reactions $n(R')$. Thus, only $n(R')$ formulas of D_{r_n} are generated. Although the size of translated formulas is enough tractable, we sometimes cannot find objective solutions since the translation is incomplete. To generate more solutions, we here extend D_{r_n} to $D^k_{r_n}$. Let o be an integer such that $o = n(R') * (k-1) + f_u(r_n)$. We then have the following formula $D^k_{r(n)}$:

$$D^k_{r(n)} = \left(rt_{n,o} \rightarrow \bigwedge_{m_i \in s(r_n)} mt_{i,o-1} \wedge \bigwedge_{m_j \in p(r_n)} mt_{j,o} \right) \wedge \left(\neg rt_{n,o} \rightarrow \bigwedge_{m_j \in p(r_n)} (\neg mt_{j,o-1} \rightarrow \neg mt_{j,o}) \right) \quad (5)$$

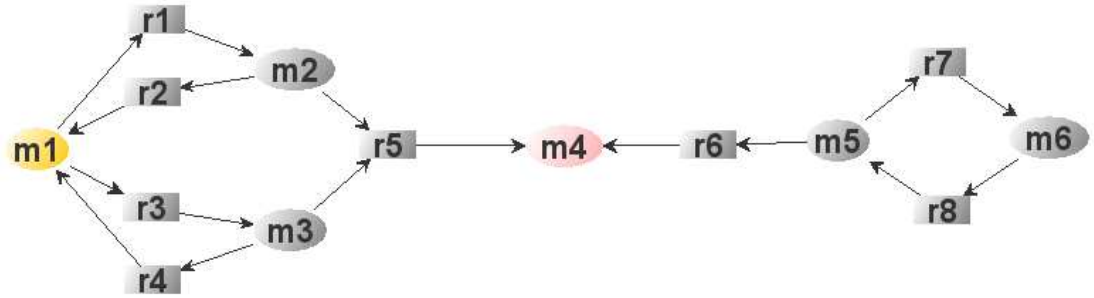

Figure 5: A Pathway Including Reversible Reactions

Obviously, D_{r_n} corresponds to $D^k_{r_n}$ when $k = 1$. Let z be an integer representing *step*. We have the following formula:

$$\bigwedge_{k=1}^{z} D^k_{r_n}. \tag{6}$$

In practice, $z = 3$ is enough to obtain the objective sub-pathways for the benchmark we used this time. We also need to have an initial condition and a target condition:

$$C(0) = \bigwedge_{m_i \in M_s \cup M_i} mt_{i,0} \wedge \bigwedge_{m_j \in M \setminus (M_s \cup M_i)} \neg mt_{j,0} \tag{7}$$

$$C(n(R') * z) = \bigwedge_{m_i \in M_t} mt_{i,n(R')*z} \tag{8}$$

Finally, we have the translated formula as follows:

$$\Psi = \bigwedge_{k=1}^{z} D^k_{r_n} \wedge C(0) \wedge C(n(R') * z) \wedge \psi_s \tag{9}$$

We use a set of propositional variables $V_p = \{mt_{i,n(R')*z} | m_i \in M\} \cup \{rt_{j,n(R')*z} | r_j \in R'\}$ to be minimized. Although we can restrict a number of solutions using a step z, sometimes there is a case that we want to reduce more number of solutions. In this case, we can choose more essential solutions by setting $V_p = \{mt_{i,n(R')*z} | m_i \in M\}$. We then compute minimal models of the formula Ψ with respect to V_p by the minimal model generation procedure shown in Figure 1.

4.2 Treate Reversible Reactions

Treatment of reversible reactions frequently becomes a problem. Ray *et al.* reported the difficulty of that and answer set semantics is suitable to resolve the problem [13]. Some other approaches took pre-processing or post-processing which breaks reversible reactions in a pathway [1, 11, 16]. Unlike those approaches, our method resolve the problem by considering the notion of activatable and finding minimal models of translated formulas. We show an toy example including reversible reactions in Figure 5. Metabolites $m1$ and $m4$ are a source metabolite and a target metabolite, respectively. We represent a model in a set of reactions to simplify explanations. A set of reactions $\{r_6, r_7, r_8\}$ cannot be a model

of translated formula due to the formula (3). The formula (3) traces the origin of the producibility of the metabolite as well as state maintenance, that is, if a metabolite is producible at a time $t = e$ then the formula (3) guarantees either the metabolite is producible at a time $t < e$ or the reaction is activatable at a time $t = e$. Therefore reversible reactions without feeding from $M_s \cup M_i$ are not activatable. Practically, such reactions are omitted in the first place by the procedure shown in Figure 3. A set of reactions $\{r_1, r_2, r_3, r_4, r_5\}$ can be a model including reversible reactions. However it cannot be a minimal model because there is a model $\{r_1, r_3, r_5\}$. Finally, we found only one minimal model $\{r_1, r_3, r_5\}$ for the example.

4.3 Other Biological Applications

Simulating Effects of Deletion of Enzymes. The method allow us to simulate the difference between pathways of wild-type organisms and pathways of mutants or gene knockout organisms. For instance, we can obtain the effect of a gene knock out by removing the reaction related to the gene we want to omit. This is achieved by adding the following formula.

$$\neg rt_{i,n(R')*z} \tag{10}$$

Simulating Effects of Inhibition. In metabolic pathways, each reaction is catalyzed by enzymes. Inhibition relations in some enzymes have been studied through biological experiments. Our method is capable to treat this relation by adding the following formula:

$$\neg rt_{i,n(R')*z} \vee \neg rt_{j,n(R')*z} \tag{11}$$

where reactions r_i and r_j are catalyzed by inhibited enzymes, respectively. This inhibition relations refine output sub-pathways of the method.

Forbidden Metabolites. A further potential application is in drug design, which restricts bi-products by the effect of compounds included in the drug. In this case, we can test by adding drug compounds as sources and unexpected bi-products as forbidden metabolites. This is achieved by adding the following formulas.

$$\bigwedge_{m_i \in M_f} \neg mt_{i,n(R')*z} \tag{12}$$

where M_f is a set of metabolites which are forbidden to present. Those constraints are useful to refine outputs when we know such forbidden metabolites in advance.

5 Experiments and Results

To evaluate our method, we use two reaction databases of *E. coli* K-12. One is the the reaction database from supplemental data of the literature [1]. Another one is from a well-known biological database *EcoCyc* [5] which gathers results of biological experiments and existence knowledge of *E. coli*. We downloaded the latest version 13.6 of the reaction database released on the November 2009. In the following experiments, we use experimentally elucidated sub-pathways as right solutions, which are respectively obtained from the literature [1] and the database *EcoCyc*. We modified the Main class

Table 1: Results for Pathways from [1]

Pathway#	Proposal			Beasley [1]		Planes [11]
	#Steps	#Sols.	cors.	cors. (a)	cors. (b)	cors.
1	3	1	yes	yes	no	no
2	1	1	yes	yes	no	yes
3	2	38	yes	yes	yes	no
4	1	1	yes	yes	no	no
5	3	4	yes	no	no	yes
6	2	7	yes	yes	no	yes
7	1	1	yes	yes	no	yes
8	3	28	yes	no	yes	no
9	1	3	yes	yes	no	yes
10	1	1	yes	yes	no	yes
Total # of yes in cors.			10	8	2	6

of the SAT solver *Minisat*[1] [6] and used it as a minimal model generator shown in Section 2. Each experiment has been done using a PC (Intel Centrino 1.84GHz CPU and 1GB RAM) running Ubuntu Linux 9.04 within 10 seconds. We have developed a graphical user interface integrating the proposed method, which aims for smooth evaluation. Figures 6 and 7 are screen shots of our experimental results from the interface.

5.1 Comparison with Previous Methods

There are two previous method. One is a method using optimization modeling for pathway analyses [1]. The input of this method is a reaction database with stoichiometry. Another one is a constraint based method for path finding [11]. The input of this method is a reaction database without stoichiometry as same as the proposed method. Due to the differences of each input, problem formalization and the number of solutions, it is difficult to make a direct comparison. We thus give an approximate comparison for 10 pathways which are used for both two methods. We use same source, initial, and target metabolites according to the literature [1]. As right solutions, the method by [11] used liner paths which are chosen from the experimentally elucidated sub-pathways from the supplemental data of the literature [1]. Similarly, we used those sub-pathways omitted bypass reactions as right solutions.

The results are shown in the table 1. First column shows the following pathways: #1 gluconeogenesis, #2 glycogen, #3 glycolysis, #4 proline bio-synthesis, #5 ketogluconate metabolism, #6 pentose phosphate, #7 salvage pathway deoxythymidine phosphate, #8 Kreb's cycle, #9 NAD biosynthesis, #10 arginine biosynthesis. Second column shows the number of steps where the most accurate sub-pathway was found. Third column shows the number of solutions found in the steps shown in the second column. Columns 4-7 show whether each method could find the structure exactly corresponding to the experimentally elucidated sub-pathways. In columns 5 and 6, (a) represents the objective of minimizing the total number of reactions and (b) represents the objective of maximizing the production of ATP in the literature [1]. As a result, we found every sub-pathway corresponding to the experimentally elucidated sub-pathways with the step $z \leq 3$. Moreover, the number of solutions are less than 10 except the pathway #3 and #8. Even for these two pathways, the number of solutions is less than 40, which is still tractable.

[1] `http://minisat.se/`

Table 2: Found Minimal Reaction Sets

Reaction Name (by [5])	M1	M2	M3	M4	M5	EcoCyc
2PGADEHYDRAT-RXN				x	x	x
3PGAREARR-RXN				x	x	x
6PFRUCTPHOS-RXN		x			x	x
6PGLUCONOLACT-RXN			x			
DLACTDEHYDROGNAD-RXN	x	x				
F16ALDOLASE-RXN		x			x	x
F16BDEPHOS-RXN						x
GAPOXNPHOSPHN-RXN				x	x	x
GLU6PDEHYDROG-RXN			x			
GLYOXIII-RXN	x	x				
KDPGALDOL-RXN			x			
METHGLYSYN-RXN	x	x				
NAD-KIN-RXN			x			
PEPDEPHOS-RXN				x	x	x
PEPSYNTH-RXN						x
PGLUCISOM-RXN	x	x		x	x	x
PGLUCONDEHYDRAT-RXN			x			
PHOSGLYPHOS-RXN				x	x	x
RXN0-313	x			x		
TRIOSEPISOMERIZATION-RXN	x					x

5.2 Accuracy Evaluation on the *E. coli* Metabolic Pathway

We also apply our method to a whole metabolic pathway of ***E. coli*** (see Figure 6). In this experiment, we choose initial metabolites by calculating percentage of the presence of each metabolites as same as the literature [1]. For instance, in generally, a cell contains 60 percents water. In order to decide such initial metabolites we define the percentage of the presence of a metabolite $pr_m = (n_m \div n(R)) \times 100$, where n_m represents the number of reactions in which a metabolite m appears. We compute this percentage of the presence for every metabolite and if it has the presence more than 1.5 percent we use it as an initial metabolite. Although 1.5 percent is apparently small value, this is relatively high presence in the pathway because there are only 16 of 1073 metabolites which have the percentage of presence more than 1.5 percent. We particularly choose metabolites which are the first 6 of 1073 metabolites regarding its percentage of presence: `WATER`, `PROTON`, `ATP`, `ADP`, `|pi|` and `NAD`. We apply the method to find a glycolysis sub-pathway in a whole ***E. coli*** pathway. As a result, we found 5 sets of reactions (see Table 2). One of them called `M5` includes 8 reactions (see Figure 7) corresponding to the conventional glycolysis sub-pathway described in EcoCyc. We evaluate the obtained sub-pathway `M5` with the following evaluation value. True positive (TP) is a number of reactions found in the experimentally elucidated solution which are also part of a computational result. False positive (FP) is a number of reactions found in a computational result but not in the experimentally elucidated solution. False negative (FN) is a number of reactions found in the experimentally elucidated solution but not in a computational result. Here we define sensitivity $Sn = TP/(TP+FN)$, positive predictive value $PPV = TP/(TP+FP)$ and accuracy $Ac = (Sn+PPV)/2$. As a result, we obtain $Sn = 0.727$, $PPV = 1$, $Ac = 0.864$ for the glycolysis sub-pathway. The value $PPV = 1$ means all reactions included in an obtained sub-pathway are also included in the experimentally elucidated sub-pathway. However, $Sn = 0.727$ means that some reactions included in the experimentally elucidated sub-pathway are not included in the obtained

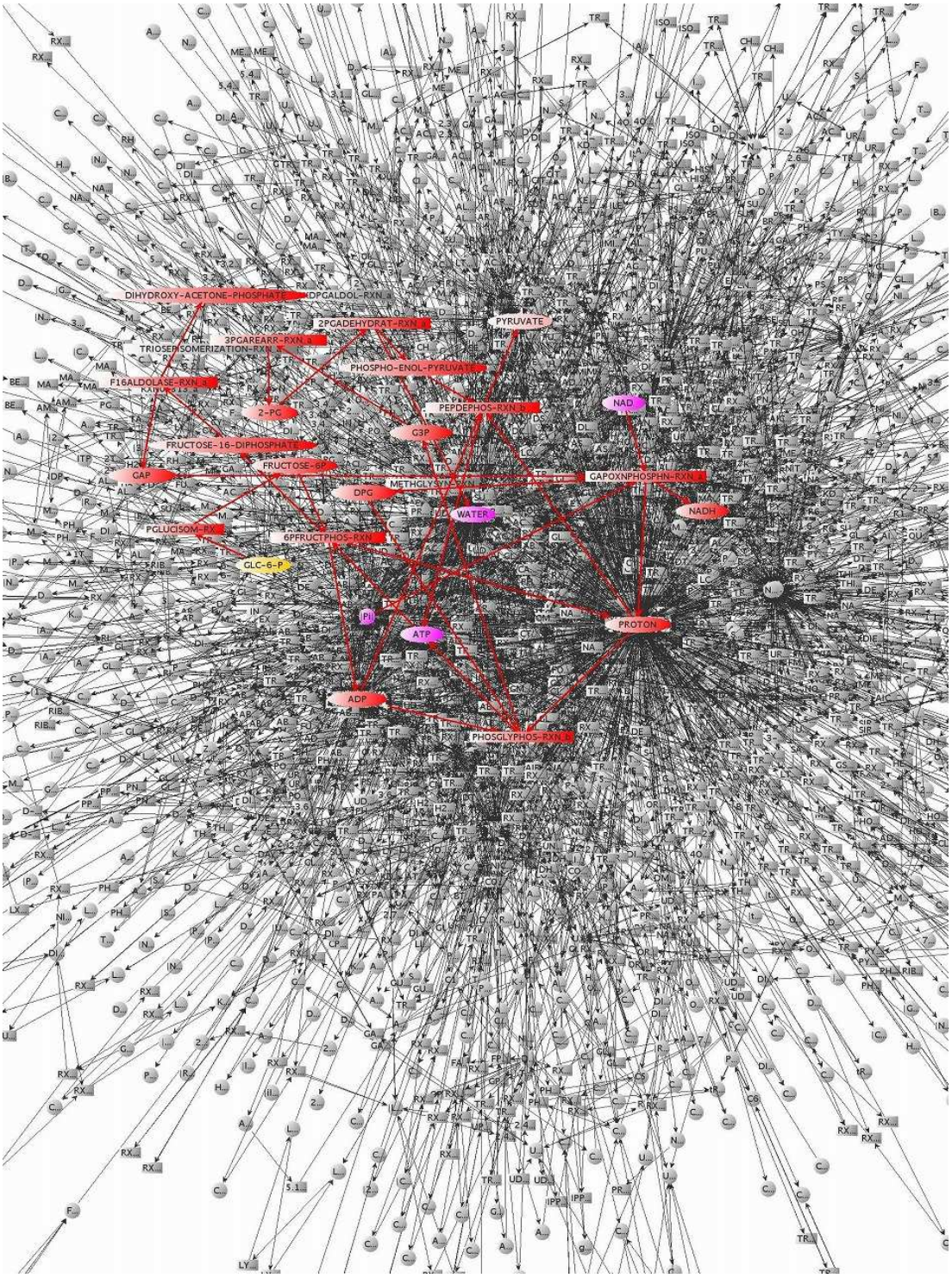

Figure 6: A Glycolysis Sub-pathway on a Whole *E. coli* Pathway

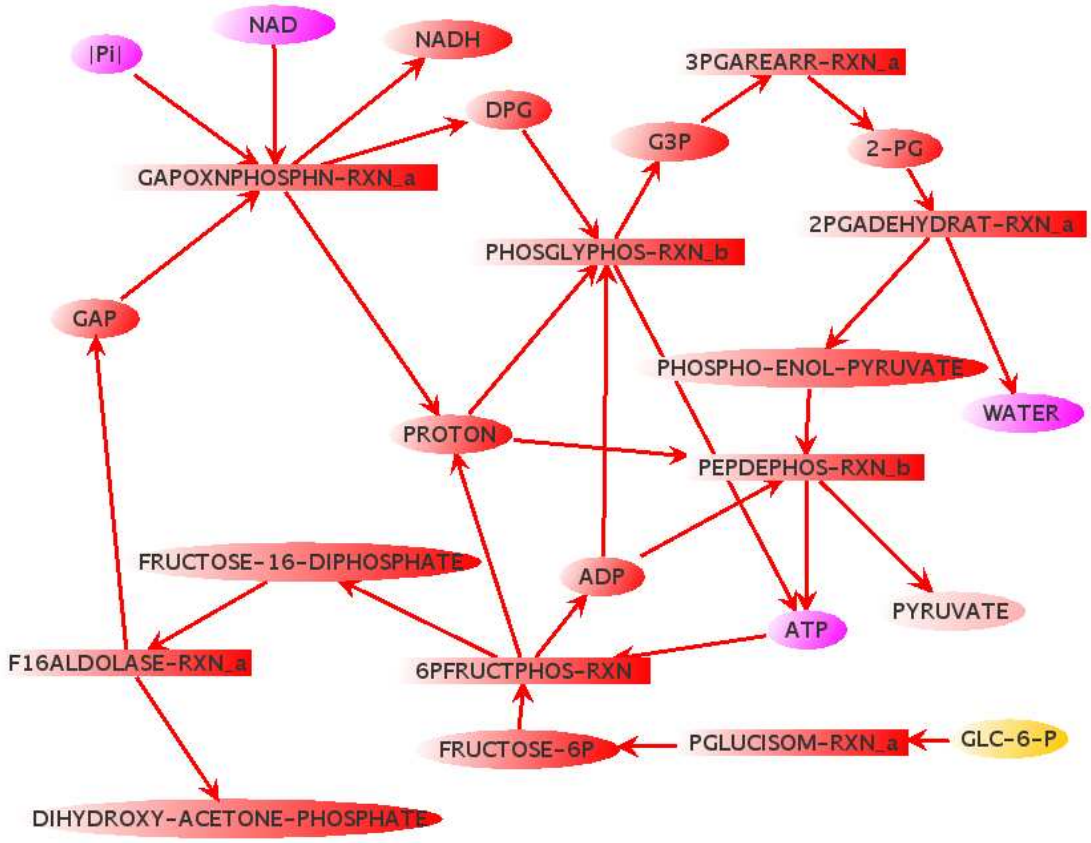

Figure 7: A Glycolysys Sub-pathway of the *E. coli* Pathway

sub-pathway. This is because the experimentally elucidated sub-pathway from EcoCyc contains bypass reactions. While the sub-pathway of EcoCyc contains a bypass reaction `PEPSYNTH-RXN` which may be needed from a stoichiometry viewpoint. In the case of the glycolysis sub-pathway, `PEPSYNTH-RXN` is such a bypass reaction. To support such bypass reactions is a future work.

6 Related Work

As far as the authors are aware, the exactly same problem of the sub-pathway finding problem has not been yet formalized. Kuffer *et al.* reported an approach via translation to petri net [9]. Although their approach considered producible and activatable, did not consider subset minimality of the solution. Schuster *et al.* proposed a concept of elementary flux modes and found minimal flux distribution [15]. Their concept of *elementary flux mode* is closed to our problem, however, they used stoichiometry information to solve their problem while our problem only consider the topology of pathways. Beasley and Planes [1] used an optimization technique to stoichiometry analyses of pathways. Although their outputs are sometimes correspond to our outputs, their optimization does not guarantee the subset minimality.

Tiwari *et al.* proposed a method which uses weighted Max-SAT solver [18]. They translated biological laws into soft constraint represented in a weighted Max-SAT problem. The method thus can

order solutions according to its total weights. However, their ordering of solutions is sometimes not acceptable from a biological viewpoint since it does not permit the activation of two reactions which uses same metabolite simultaneously. Ray *et al.* reported the logical approach for analyzing pathways using answer set programming (ASP) and reported how it suits for pathway analyzing. Similarly, Schaub and Thiele [14] apply ASP technique to analyze pathways. These approaches are also interesting in terms of translating the relations of reactions into logical form. As far as the authors know, there is a few methods have been reported for analyses of a whole organism pathway. We believe that our method provides a new linking method between simple graph-based approaches and those logical methods, which enables us to analyze complex networks like cells, organisms and life.

7 Conclusion

In this paper, we formalized the sub-pathway finding problem which identifies necessary reactions to produce target metabolites and presented a translation into a propositional formula. Our method uses a SAT solver as a model generator and it has the following features. First, our method can treat reversible reactions without pre-processing and post-processing. Second, it is capable to treat a whole *E. coli* pathway. Third, it can restrict the number of solutions to be tractable. These are important features for the realistic size pathways such as a whole cell or more extended pathways which includes metabolic, signalling, and gene regulatory networks. There are several important future topics. The proposed method found each conventional sub-pathways of the 11 pathways on *E. coli*. For more general evaluation, statistical analyses with more number of pathways are needed. We also need to consider the quality of solutions as well as ranking. Translating more biological knowledge is important to find sub-pathways of more extended pathways.

Acknowledgement This research is supported in part by the 2008-2011 JSPS Grant-in-Aid for Scientific Research (A) (No.20240016) and by the JSPS Research Fellowships for Young Scientists. We would like to thank Gauvain Bourgne and colleagues for their helpful comments. We also thank Oliver Ray for useful discussions.

References

[1] John E. Beasley and Francisco J. Planes. Recovering metabolic pathways via optimization. *Bioinformatics*, 23(1):92–98, 2007.

[2] Didier Croes, Fabian Couche, Shoshana J. Wodak, and Jacques van Helden. Metabolic pathfinding: inferring relevant pathways in biochemical networks. *Nucleic Acids Research*, 33(Web-Server-Issue):326–330, 2005.

[3] Didier Croes, Fabian Couche, Shoshana J. Wodak, and Jacques van Helden. inferring meaningful pathways in weighted metabolic networks. *Journal of Molecular Biology*, 356(1):222–236, 2006.

[4] Luis F. de Figueiredo, Stefan Schuster, Christoph Kaleta, and David A. Fell. Can sugars be produced from fatty acids? a test case for pathway analysis tools. *Bioinformatics*, 24(22):2615–2621, 2008.

[5] EcoCyc. `http://biocyc.org/download.shtml`.

[6] Niklas Eén and Niklas Sörensson. An extensible SAT-solver. In *Proceedings of SAT*, pages 502–518, 2003.

[7] Hidde De Jong. Modeling and simulation of genetic regulatory systems: A literature review. *Journal of Computational Biology*, 9:67–103, 2002.

[8] Miyuki Koshimura, Hidetomo Nabeshima, Hiroshi Fujita, and Ryuzo Hasegawa. Minimal model generation with respect to an atom set. In *Proceedings of FTP'09*, pages 49–59, 2009.

[9] Robert Küffner, Ralf Zimmer, and Thomas Lengauer. Pathway analysis in metabolic databases via differetial metabolic display (dmd). In *German Conference on Bioinformatics*, pages 141–147, 1999.

[10] I. Niemelä. A tableau calculus for minimal model reasoning. In *Proceedings of the TABLEAU '96*, pages 278–294, 1996.

[11] Francisco J. Planes and John E. Beasley. Path finding approaches and metabolic pathways. *Discrete Applied Mathematics*, 157(10):2244–2256, 2009.

[12] Syed Asad Rahman, P. Advani, R. Schunk, Rainer Schrader, and Dietmar Schomburg. Metabolic pathway analysis web service (pathway hunter tool at cubic). *Bioinformatics*, 21(7):1189–1193, 2005.

[13] Oliver Ray, Ken E. Whelan, and Ross D. King. Logic-based steady-state analysis and revision of metabolic networks with inhibition. In *CISIS*, pages 661–666, 2010.

[14] Torsten Schaub and Sven Thiele. Metabolic network expansion with answer set programming. In *ICLP '09: Proceedings of the 25th International Conference on Logic Programming*, pages 312–326, Berlin, Heidelberg, 2009. Springer-Verlag.

[15] Stefan Schuster, David A. Fell, and Thomas Dandekar. A general definition of metabolic pathways useful for systematic organization and analysis of complex metabolic networks. *Nature Biotechnology*, 18:326–332, 2000.

[16] Takeyuki Tamura, Kazuhiro Takemoto, and Tatsuya Akutsu. Measuring structural robustness of metabolic networks under a boolean model using integer programming and feedback vertex sets. In *CISIS*, pages 819–824, 2009.

[17] Marco Terzer, Nathaniel D. Maynard, Markus W. Covert, and Jörg Stelling. Genome-scale metabolit networks. *Systems Biology and Medicine*, 1(3):285 – 297, 2009.

[18] Ashish Tiwari, Carolyn L. Talcott, Merrill Knapp, Patrick Lincoln, and Keith Laderoute. Analyzing pathways using sat-based approaches. In *AB*, pages 155–169, 2007.