MetNetComp: Database for Minimal and Maximal Gene-Deletion Strategies for Growth-Coupled Production of Genome-Scale Metabolic Networks

Takeyuki Tamura

***Abstract*—Growth-coupled production, in which cell growth forces the production of target metabolites, plays an essential role in the production of substances by microorganisms. The strains are first designed using computational simulation and then validated by biological experiments. In the simulations, gene-deletion strategies are often necessary because many metabolites are not produced in the natural state of the microorganisms. However, such information is not available for many metabolites owing to the requirement of heavy computation, especially when many gene deletions are required for genome-scale models. A database for such information will be helpful. However, developing such a database is not straightforward because heavy computation and the existence of replaceable genes render difficulty in efficient enumeration. In this study, the author developed efficient methods for enumerating minimal and maximal gene-deletion strategies and a web-based database system. MetNetComp provides information on 1) a total of 85,611 gene-deletion strategies excluding apparent duplicate counting for replaceable genes for 1,735 target metabolites, 11 constraint-based models, and 10 species; 2) necessary substrates and products in the process; and 3) reaction rates that can be used for visualization. MetNetComp is helpful for strain design and for new research paradigms using machine learning.**

***Index Terms*—Biology and genetics, chemistry, combinatorial algorithms, graphs and networks, linear programming, scientific databases.**

I. Introduction

COMPUTATIONAL strain designs for microbial cell factories for substance production are evaluated in simulation before screening by biological experiments [1]. Such simulations are often conducted using *constraint-based models*, which can perform genome-scale metabolic simulations efficiently by restricting the analysis to steady states [2]. Constraint-based models are also referred to as genome-scale models.

In constraint-based models, metabolism is modeled as follows: 1) the sum of incoming reaction rates is equal to the sum of outgoing reaction rates for each metabolite, 2) the ratio of the produced and consumed metabolites in each reaction satisfies the ratio in the chemical reaction equation, and 3) each reaction rate is given upper and lower bounds.

Each constraint-based model includes a virtual reaction that represents cell growth. The cell growth reaction was designed to match the results of the biological experiments. The standard simulation in constraint-based models maximizes the cell growth rate (*GR*) because strains with high growth rates are more likely to remain in the culture during passaging. In contrast, the reaction that produces the desired metabolite is called the production reaction, and its production rate is denoted as *PR*. Therefore, the designed strains are evaluated using PR at GR maximization in the simulations (See Fig. 1(a)). The simultaneous occurrence of cell growth and target metabolite production is known as *growth-coupled production*. The design of constraint-based models plays an important role in growth-coupled production of substances through microbial metabolism [3], [4], [5], [6], [7], [8], [9], [10].

Gene-protein-reaction (GPR) rules represent relations between genes and reactions by Boolean functions whose inputs and outputs are genes and reactions, respectively. Constraint-based models can be controlled by gene deletions. If a reaction is assigned zero via gene deletions, the lower and upper bounds of the reaction rate are forced to be zero, and the reaction is deleted.

For many target metabolites, it is necessary to calculate gene-deletion strategies for growth-coupled production because only a limited number of metabolites are produced with growth coupling in the natural state of microorganisms. Many methods have been proposed to address this problem [11], [12], [13], [14], [15], [16], [17], [18], [19]. However, calculating a gene-deletion strategy that results in growth-coupled production is still challenging due to heavy computation, especially when GPR rules are complex and many gene deletions are considered for genome-scale models [20], [21], [22].

The Bigg Models database [23] contains more than 100 genome-scale constraint-based metabolic networks, many of which are used as the gold standard in computational strain design. However, information on gene-deletion strategies for growth-coupled production is unavailable. A database of such gene-deletion strategies would be helpful. However, the heavy

Manuscript received 9 March 2023; revised 25 August 2023; accepted 18 September 2023. Date of publication 22 September 2023; date of current version 26 December 2023. The work of Takeyuki Tamura was supported by JSPS, KAKENHI under Grant #20H04242.

The author is with the Bioinformatics Center, Institute for Chemical Research, Kyoto University, Uji, Kyoto 6110011, Japan (e-mail: tamura@kuicr.kyoto-u.ac.jp).

MetNetComp (https://metnetcomp.github.io/database1/indexFiles/index.html).
Digital Object Identifier 10.1109/TCBB.2023.3317837

© 2023 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

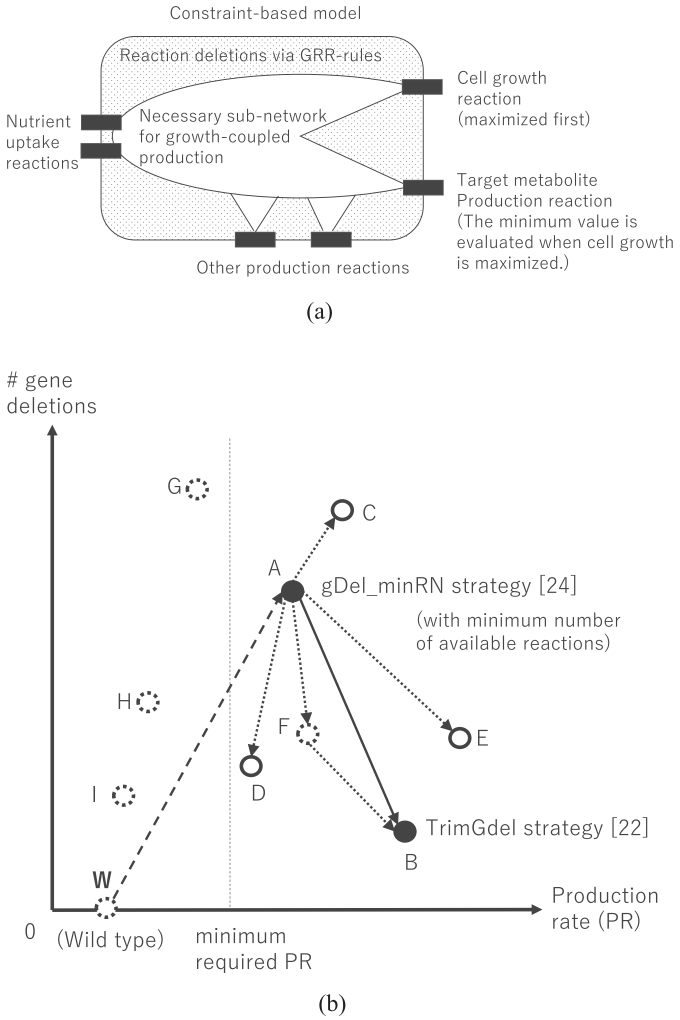

Constraint-based model
Reaction deletions via GRR-rules
Nutrient uptake reactions
Necessary sub-network for growth-coupled production
Cell growth reaction (maximized first)
Target metabolite Production reaction (The minimum value is evaluated when cell growth is maximized.)
Other production reactions
(a)
# gene deletions
G
C
A
gDel_minRN strategy [24]
(with minimum number of available reactions)
H
F
E
I
D
TrimGdel strategy [22]
B
W
0
(Wild type)
minimum required PR
Production rate (PR)
(b)

Fig. 1. (a) Illustration of gene deletions resulting in growth-coupled production, in which the target metabolite is produced when cell growth is maximized. (b) The MetNetComp database contains the maximal and minimal gene-deletion strategies for each target metabolite and the constraint-based model. Starting with gDel_minRN strategy [24], the developed algorithms AddGdel and RandTrimGdel computed maximal and minimal gene-deletion strategies such as Node C and Nodes B, E, and D, respectively.

computation for each gene-deletion strategy and combinatorial explosion caused by replaceable genes render difficulty in the efficient enumeration.

Recently, Tamura et al. developed gDel_minRN [24] to compute gene-deletion strategies that maximize the number of repressed reactions to extract the core part for growth-coupled production. Because a small number of gene deletions are preferable in traditional metabolic engineering, the author extended gDel_minRN to TrimGdel [22] to compute gene-deletion strategies in which the number of gene deletions is minimal.

Fig. 1(b) illustrates the relation between gene-deletion strategies obtained by gDel_minRN and TrimGdel. The horizontal and vertical axes indicate the PR and number of gene deletions, respectively. The black circles A and B indicate the gDel_minRN and TrimGdel strategies, respectively. Arrows on the upper and lower sides represent gene deletions and additions, respectively. Because B is obtained by gene additions, that is, trimming gene deletions, from A, there is an edge from A to B. A dashed circle W indicates the wild type, where no gene is deleted, and the minimum required PR is not satisfied.

During the enumeration of gene-deletion strategies, it is sufficient to show minimal and maximal gene-deletion strategies instead of listing all of them. For example, in the arrow between A and B, gene-deletion strategies located in the arrow always result in growth-coupled production. Suppose that the difference in the number of gene deletions between A and B is k. Even when only A and B are given, it is sufficient to examine $O(k^2)$ strategies to find all the gene-deletion strategies associated with the arrow between A and B. When a gene-deletion strategy is provided, a polynomial time algorithm is available for checking whether it results in growth-coupled production because linear programming is solvable in polynomial time [25]. Therefore, it is sufficient to demonstrate only A and B.

In this study, the author developed the MetNetComp database that enumerates the minimal and maximal gene-deletion strategies for growth-coupled production. To compute such gene-deletion strategies, the author first developed two algorithms: AddGdel and RandTrimGdel. AddGdel deletes more genes from the gDel_minRN strategy to obtain maximal gene-deletion strategies. For example, in Fig. 1(b), AddGdel finds C by deleting genes from the gDel_minRN strategy A. RandTrimGdel repeatedly employs an extended version of TrimGdel [22], which computes one of the minimal gene-deletion strategies by trimming gene-deletion strategies obtained by gDel_minRN. Minimal gene-deletion strategies for growth-coupled production can be enumerated by applying RandTrimGdel to the gDel_minRN strategy,

For example, in Fig. 1(b), RandTrimGdel finds D and E in addition to B. RandTrimGdel classifies genes into representative and alternative genes to avoid apparent duplicate counting by apparently replaceable genes. RandTrimGdel focuses on representative genes to reduce the number of case separations and improve enumeration efficiency.

MetNetComp contains 85,611 gene-deletion strategies for 1,735 target metabolites, 10 species, and 11 constraint-based models, which are listed as gold-standard models in the Bigg Models database. When a target metabolite and constraint-based model are designated, users can download a comma-separated value (CSV) file that shows the list of gene-deletion strategies and their resulting GR and PR. The CSV files are viewed in Excel. In the CSV file, the gene-deletion strategies are sorted in ascending order of the number of deleted genes. If the number of deleted genes is the same, they are sorted by PR. Each gene-deletion strategy has a corresponding web page that shows the GR, PR, uptaken metabolites (nutrients), and (target and non-target) products. Users can also download a JavaScript Object Notation (JSON) file that contains information on each reaction rate of the resulting metabolic flow. The JSON files for e_coli_core, iML1515, and iMM904 can be applied to Escher [26], which visualizes the metabolic flow. If a user first selects a target metabolite, MetNetComp shows the list of constraint-based models that can produce it with growth-coupling. In contrast, if a user first selects a constraint-based model, MetNetComp shows the list of target metabolites for which gene-deletion strategies are available. Metabolites and reactions in MetNetComp are linked to the Bigg Models database, whereas genes are linked to the Kyoto Encyclopedia of Genes and Genomes (KEGG) database [27]. The CSV and JSON files can be systematically downloaded using the URLs illustrated in Section III.

Because microbial cell factories require numerous computational strain designs before screening by in vivo experiments [1], the mass data on growth-coupling gene-deletion strategies provided by MetNetComp are expected to be helpful. Moreover, such mass data may lead to the development of new research paradigms based on machine learning, for instance, a study to predict the genes that should be deleted with the topology of the metabolic network and the structure of the target metabolite.

The remainder of this paper is organized as follows: Section II-A describes the main problem of this study mathematically; Sections II-B and II-C describe the developed algorithms RandTrimGdel and AddGdel mathematically, respectively; Section II-D illustrates the main problem, the behavior of AddGdel and RandTrimGdel, and MetNetComp using toy examples; Section III describes details of the MetNetComp database; Section IV evaluates the MetNetComp database and developed methods and discusses the scope of the future studies.

II. Methods

A. Definition

The *constraint-based metabolic network* with *GPR rules* is defined as $C = (C_1, C_2) = (M, R, S, L, U, G, F, P)$. $C_1 = (M, R, S, L, U)$ denotes a constraint-based metabolic network. $M = \{m_1, \ldots, m_a\}$ and $R = \{r_1, \ldots, r_b\}$ are sets of metabolites and reactions, respectively. S is a stoichiometric matrix, where $S_{ij} = k$ implies that r_j produces k of m_i per unit time. If k is a negative number, then m_i is consumed. If $S_{i,j}$ is all positive or negative for a fixed j, r_j is called an *external* reaction. Let R_{ext} be the set of external reactions. Let $V = \{v_1, \ldots, v_b\}$ be a set of reaction rates per unit time (flux) of R. Let $L = \{l_1, \ldots, l_b\}$ and $U = \{u_1, \ldots, u_b\}$ be the sets of lower and upper bounds of the reaction rates (speed) for V, respectively. R always includes the *growth reaction* r_{growth}, which is a special virtual reaction representing cell growth. The cell growth flux is represented by v_{growth}, which is called *growth rate* (*GR*). Let m_t be the target metabolite that must be produced. C_1 includes at most one *production reaction* $r_{production}$ that produces m_t. Let $v_{production}$ be the *production rate* (*PR*). If C_1 does not initially have $r_{production}$, then a virtual reaction $r_{production}$ is added to evaluate the PR. In this study, PR was defined as the minimum $v_{production}$ when the cell growth rate v_{growth} was maximized.

$C_2 = (G, F, P)$ is the *GPR rule* for C. $G = \{g_1, \ldots, g_c\}$ and $F = \{f_1, \ldots, f_b\}$ are sets of genes and Boolean functions, respectively. $P = \{p_1, \ldots, p_b\}$ is the set of the outputs of F.

If $p_j = 0$, then l_j and u_j are forced to zero, and r_j is repressed. If every p is identical when g_i and g_j $(i < j)$ are exchanged, then g_j is called an *alternative* gene for g_i. Let us state g_j is a *representative* gene if g_j is not an alternative gene for any g_i $(i < j)$. Let G^R be the set of representative genes. $G^A = G - G^R$ denotes a set of alternative genes. If $g_i \in G^R$ is representative of an alternative gene $g_j \in G^A$, let $Rep(g_j) = g_i$.

In the standard setting of growth-coupling simulation using flux balance analysis (FBA) [2] with C_1 and C_2, PR is evaluated when GR is maximized by the following linear programming (LP):

$$
\begin{array}{l}
\textbf{maximize} \\
\quad v_{growth} \ (= \text{GR}) \\
\textbf{such that} \\
\quad \Sigma_j S_{ij} v_j = 0 \text{ for all } i \\
\quad p_j = f_j(G) \text{ for all } j \\
\quad \begin{cases} v_j = 0 & \text{when} \quad p_j = 0, \\ l_j \leq v_j \leq u_j & \text{when} \quad p_j = 1 \end{cases} \\
\quad i = \{1, \ldots, a\}, j = \{1, \ldots, b\} \\
\quad p, g \in \{0, 1\} \\
\quad (\text{evaluate PR} = \textbf{min } v_{production})
\end{array}
$$

Because LP does not always have a unique solution, the minimum PR at GR maximization is evaluated in this study.

Let D be a set of genes assigned zero in G. Let D^R and D^A be the sets of representative and alternative genes assigned to zero in G^R and G^A, respectively. Genes assigned to zero are called *deleted* genes, whereas genes assigned to one are called *remaining* genes. When a 0/1 assignment for G induced by D results in $v_{growth} \geq GRLB$ and $v_{production} \geq PRLB$ during the aforementioned formalization, where GRLB and PRLB are the minimum required GR and PR, respectively, let D be called a *growth-coupling gene-deletion strategy*. GRLB = PRLB = 0.001 was applied in this study.

If the number of deleted genes is minimal for a growth-coupling gene-deletion strategy, G is called a *minimal gene-deletion strategy*. Hence, for a minimal gene-deletion strategy G, if a deleted gene is changed to a remaining gene, G will not achieve growth-coupled production. Similarly, if the number of deleted genes is maximal for a growth-coupling gene-deletion strategy, G is called a *maximal gene-deletion strategy*.

The MetNetComp database provides information on the minimal and maximal growth-coupling gene-deletion strategies when C and m_t are designated. If only one of C and m_t is designated, then the elements of the other with growth-coupling gene-deletion strategies are listed. For each growth-coupling gene-deletion strategy, MetNetComp displays information on v_i for $r_i \in R_{ext}$. A list of all $v \in V$ is also available.

B. Algorithm and Database for Minimal Gene-Deletion Strategies

The pseudo-codes of RandTrimGdel and its sub-modules GRPRchecker and DeletionShift are outlined as follows.

```
Procedure RandTrimGdel($C$, $m_t$, max_$loop$)
  /*Step 1*/
  [$D$, $GR_{threshold}$, $PR_{threshold}$]
        = gDel_minRN($C$, $m_t$, max_$loop$)
```

```
Classify $D$ into $D^R$ and $D^A$
$[D^R, D^A]$ =DeletionShift$(C, D^R, D^A)$
/*Step 2*/
for $loop = 1$ to $\max\_loop$
  $i = 1, j = 1$
  while $i \le |D^R|$
    if $i = 1$
      /*random permutation of deleted gene
      indexes*/
      $D^R = \{g^0_{\sigma(1)}, \ldots, g^0_{\sigma(|D^R|)}\}$
    end
    $D^R = D^R - g^0_i$
    $result =$ GRPRchecker$(C, m_t, D^R \cup D^A,$
        $GR_{threshold}, PR_{threshold})$
    if $result = success$
      $i = 1$
    else
      $D^R = D^R \cup g^0_i$
      $i = i + 1$
    end
    $[D^R, D^A] =$ DeletionShift$(C, D^R, D^A)$
  end
  if $D_k \nsubseteq D^R \cup D^A$ for all $1 \le k < j$ then
    $D_j = D^R \cup D^A$
    $j = j + 1$
  end
end
return $\mathbf{D} = \{D_1,..., D_{j-1}\}$
```

```
$[result] =$ GRPRchecker$(C, m_t, D, GR_{threshold},$
 $PR_{threshold})$
  maximize
    $v_{growth}$
  such that
    $\Sigma_j S_{ij} v_j = 0$ for all $i$
    $\{$ $v_j = 0$ if $p_j = 0$
       $l_j \le v_j \le u_j$, otherwise
    $p_j = f_j(G)$
    $\{$ $g = 0$,  if  $g \in D$
       $g = 1$,  otherwise
  if $v_{growth} \ge GR_{threshold}$
            and (min $v_{production}) \ge PR_{threshold}$
    return $success$
  else
    return $false$
  end
```

```
$[D^R, D^A] =$ DeletionShift$(C, D^R, D^A)$
  forall $g \in |D^A|$
    if $Rep(g) \notin D^R$ then
      $D^R = D^R \cup g$, $D^A = D^A - g$
    end
  end
  return $D^R, D^A$
```

For C and m_t, the MetNetComp database displays a set of minimal gene-deletion strategies $\mathbf{D} = \{D_1, \ldots, D_{j-1}\}$ and a set of alternative genes for each gene in D.

C. Algorithm for Maximal Gene-Deletion Strategies

AddGdel aims to determine a maximal gene-deletion strategy $G_{\max}$ for a given constraint-based model C and target metabolite m_t.

The pseudo-codes of AddGdel and its sub-module RemainShift are as follows.

```
Procedure AddGdel$(C, m_t)$
  /*Step 1*/
  $[D, GR_{threshold}, PR_{threshold}]$
        $=$ gDel_minRN$(C, m_t, \max\_loop)$
  /*Step 2*/
  $G_{rem} = \{g^1_1, \ldots, g^1_{|G-D|}\}$ /*a set of remaining
   genes*/
  $i = 1$
  while $i \le |G_{rem}|$
    $D = D \cup g^1_i$
    $result$ = GRPRchecker$(C, m_t, D, GR_{threshold},$
    $PR_{threshold})$
    if $result = success$ then
      $i = 1$
      $G_{rem} = \{g^1_1, \ldots, g^1_{|G-D|}\}$
    else
      $D = D - g^1_i$
      $i = i + 1$
    end
  end
  $[D^R, D^A]$ = RemainShift$(C, D^R, D^A)$
  return $G_{rem} = G - D$
```

```
$[D^R, D^A] =$ RemainShift$(C, D^R, D^A)$
  forall $g^R \in |D^R|$
    if $g^R = Rep(g^A)$ for some $g^A \notin D^A$ then
      (If there are multiple such $g^A$,
        the one with the smallest index is selected.)
      $D^R = D^R - g^R$, $D^A = D^A \cup g^A$
    end
  end
  return $D^R, D^A$
```

For C and m_t, the MetNetComp database displays a set of remaining genes G_{rem} that corresponds to the maximal gene-deletion strategy and a set of alternative genes for each gene in G_{rem}.

D. Example

1) Constraint-Based Metabolic Network With GPR Rules: Fig. 2 shows a toy example of the constraint-based metabolic network $C = (C_1, C_2) = (M, R, S, L, U, G, F, P)$.

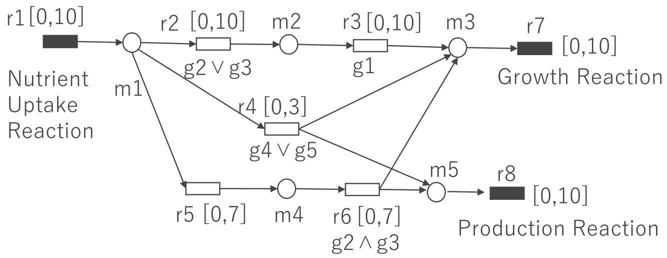


Fig. 2. Small example of a constraint-based metabolic network with GPR rules. Section II-D illustrates the main problem and the developed algorithms RandTrimGdel and AddGdel.

$M = \{m_1, \ldots, m_5\}$ and $R = \{r_1, \ldots, r_8\}$ denote sets of metabolites and reactions, respectively. The target metabolite m_t is m_5. The growth and target metabolite production reactions are r_7 and r_8, respectively. The stoichiometric matrix is

$$S = \begin{pmatrix} 1 & -1 & 0 & -1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 \end{pmatrix}.$$

$[x, y]$ denotes the lower and upper bounds for each reaction rate, that is, $L = (0, 0, 0, 0, 0, 0, 0, 0)$ and $U = (10, 10, 10, 3, 7, 7, 10, 10)$.

The set of genes, Boolean functions, and outputs of Boolean functions are $G = \{g_1, \ldots, g_5\}$, $F = \{f_1, \ldots, f_8\}$, and $P = \{p_1, \ldots, p_8\}$, respectively, where

$$\begin{aligned} f_1 &: p_1 = 1, \\ f_2 &: p_2 = g_2 \vee g_3, \\ f_3 &: p_3 = g_1, \\ f_4 &: p_4 = g_4 \vee g_5, \\ f_5 &: p_5 = 1, \\ f_6 &: p_6 = g_2 \wedge g_3, \\ f_7 &: p_7 = 1, \\ f_8 &: p_8 = 1. \end{aligned}$$

For example, since the GPR rule for r_2 is given as $f_2 : p_2 = g_2 \vee g_3$, the reaction rate of r_2, denoted by v_2, is forced to be zero if both g_2 and g_3 are zero, whereas $0 \leq v_2 \leq 10$ is held if either g_2 or g_3 is one. However, v_6 is forced to be zero if either g_2 or g_3 is zero, whereas $0 \leq v_6 \leq 7$ is held if both g_2 and g_3 are one. Since $p_1 = p_5 = p_7 = p_8 = 1$ always holds, none of v_1, v_5, v_7, v_8 are forced to be zero by genes.

2) Behavior of Constraint-Based Model: Table I(a) describes the patterns of the gene deletions in this example: $2^5 = 32$ patterns are classified into eight cases according to the flux distributions. In its original state, that is, when no gene is deleted, the maximum value of GR is $v_7 = 10$. However, there are three paths to reach from the nutrient uptake reaction r_1 to the growth reaction r_7: (r_1, r_2, r_3, r_7), (r_1, r_4, r_7), (r_1, r_5, r_6, r_7). If the first path is not used, GR = 10 and PR = 10 are obtained, as shown in ID 1 in Table I(b). This is the most optimistic case regarding the value of PR. However, if only the first path is used, GR = 10 and PR = 0 are obtained as shown in ID 2 of Table I(a). This is the most pessimistic case regarding the value of PR. In this study, we evaluate the most pessimistic value of PR when the GR is maximized. Therefore, GR = 10 and PR = 0 are obtained in the original state. Gene-deletion strategies $\{g_4\}$ and $\{g_5\}$ are also classified as Type 1 because the same flux distribution is obtained.

TABLE I
(A) GENE-DELETION STRATEGIES FOR THE EXAMPLE IN FIG. 2 ARE CLASSIFIED INTO EIGHT TYPES ACCORDING TO THE RESULTING FLUX DISTRIBUTIONS (REACTION RATES). (B) THE RESULTING FLUX DISTRIBUTION OF TYPES 1 TO 8 GENE-DELETION STRATEGIES. FOR EACH TYPE, OPTIMISTIC AND PESSIMISTIC PRODUCTION RATES (PR) AT GROWTH RATE (GR) MAXIMIZATION ARE SHOWN

Type	Gene KO classification with regard to flux distribution
1	ϕ, $\{g_4\}$, $\{g_5\}$
2	$\{g_1\}$, $\{g_1, g_4\}$, $\{g_1, g_5\}$
3	$\{g_2\}$, $\{g_3\}$, $\{g_2, g_4\}$, $\{g_2, g_5\}$, $\{g_3, g_4\}$, $\{g_3, g_5\}$
4	$\{g_1, g_2\}$, $\{g_1, g_3\}$, $\{g_2, g_3\}$, $\{g_1, g_2, g_3\}$, $\{g_1, g_2, g_4\}$, $\{g_1, g_2, g_5\}$, $\{g_1, g_3, g_4\}$, $\{g_1, g_3, g_5\}$, $\{g_2, g_3, g_4\}$, $\{g_2, g_3, g_5\}$, $\{g_1, g_2, g_3, g_4\}$, $\{g_1, g_2, g_3, g_5\}$
5	$\{g_4, g_5\}$
6	$\{g_1, g_4, g_5\}$
7	$\{g_2, g_4, g_5\}$, $\{g_3, g_4, g_5\}$
8	$\{g_1, g_2, g_4, g_5\}$, $\{g_1, g_3, g_4, g_5\}$, $\{g_2, g_3, g_4, g_5\}$, $\{g_1, g_2, g_3, g_4, g_5\}$

(a)

ID	Gene KO		v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
1	Type 1	best	10	0	0	3	7	7	10	10
2		worst	10	10	10	0	0	0	10	0
3	Type 2	best	10	0	0	3	7	7	10	10
4		worst	10	0	0	3	7	7	10	10
5	Type 3	best	10	7	7	3	0	0	10	3
6		worst	10	10	10	0	0	0	10	0
7	Type 4	best	3	0	0	3	0	0	3	3
8		worst	3	0	0	3	0	0	3	3
9	Type 5	best	10	3	3	0	7	7	10	7
10		worst	10	10	10	0	0	0	10	0
11	Type 6	best	7	0	0	0	7	7	7	7
12		worst	7	0	0	0	7	7	7	7
13	Type 7	best	10	10	10	0	0	0	10	0
14		worst	10	10	10	0	0	0	10	0
15	Type 8	best	0	0	0	0	0	0	0	0
16		worst	0	0	0	0	0	0	0	0

(b)

IDs 3 and 4 of Table I(b) describe the optimistic and pessimistic flux distributions regarding the PR value when g_1 is deleted. When g_1 is deleted, $f_3 : p_3 = g_1 = 0$ is obtained and v_3 is forced to be zero. The maximum GR is still $v_7 = 10$ using the second and third paths of r_1 to r_7, the flux distribution is uniquely determined, $v_4 = 3$, $v_5 = v_6 = 7$, $v_7 = v_8 = 10$ are obtained, and GR and PR are 10 for both optimistic and pessimistic cases regarding the PR value. The gene-deletion strategies $\{g_1, g_4\}$ and $\{g_1, g_5\}$ are also classified as Type 2 as shown in Table I(a).

IDs 5 and 6 of Table I(b) describe the flux distributions when g_2 is deleted: $p_6 = g_2 \wedge g3 = 0$ is obtained and v_6 is forced to be zero. The maximum GR is $v_7 = 10$. In the optimistic case, the second path is fully used, and GR = 10 and PR = 3 are obtained. In the pessimistic case, only the first path is used, and GR $= 10$ and PR $= 0$ are obtained. Gene-deletion strategies $\{g_3\}, \{g_2, g_4\}, \{g_2, g_5\}, \{g_3, g_4\}, \{g_3, g_5\}$ are also classified as Type 3 as shown in Table I(a).

IDs 7 and 8 of Table I(b) describe the flux distributions when $\{g_1, g_2\}$ is deleted: $p_3 = g_1 = 0$ and $p_6 = g_2 \wedge g3 = 0$ are obtained, and v_3 and v_6 are forced to be zero. Because only the second path can be used, the maximum GR is $v_7 = 3$, and the PR is uniquely determined as $v_8 = 3$. Then, GR $=$ PR $=$ 3 is obtained for both optimistic and pessimistic cases. Gene-deletion strategies $\{g_1, g_3\}$, $\{g_2, g_3\}$, $\{g_1, g_2, g_3\}$, $\{g_1, g_2, g_4\}$, $\{g_1, g_2, g_5\}$, $\{g_1, g_3, g_4\}$, $\{g_1, g_3, g_5\}$, $\{g_2, g_3, g_4\}$, $\{g_2, g_3, g_5\}$, $\{g_1, g_2, g_3, g_4\}$, and $\{g_1, g_2, g_3, g_5\}$ are also classified as Type 4, as shown in Table I(a).

IDs 9 and 10 of Table I(b) describe the flux distributions when $\{g_4, g_5\}$ is deleted: $p_4 = g_4 \vee g_5 = 0$ is obtained, v_4 is forced to be 0, and the second path cannot be used. In the optimistic case, the third path is fully used, and GR $= 10$ and PR $= 7$ are obtained. In the pessimistic case, only the first path is used, and GR $= 10$ and PR $= 0$ are obtained.

IDs 11 and 12 of Table I(b) describe the flux distributions when $\{g_1, g_4, g_5\}$ is deleted: $p_3 = g_1 = 0$ and $p_4 = g_4 \vee g_5 = 0$ are obtained, v_3 and v_4 are forced to be 0, and only the third path can be used. Subsequently, the flux distribution is uniquely determined, and GR $=$ PR $= 7$ is obtained.

IDs 13 and 14 of Table I(b) describe the flux distributions when $\{g_2, g_4, g_5\}$ is deleted: $p_4 = g_4 \vee g_5 = 0$ and $p_6 = g_2 \wedge g_3 = 0$ are obtained, v_4 and v_6 are forced to be 0, and only the first path can be used. Subsequently, the flux distribution is uniquely determined, and GR $= 10$ and PR $= 0$ are obtained. The gene-deletion strategy $\{g_3, g_4, g_5\}$ is also classified as Type 7 as shown in Table I(a).

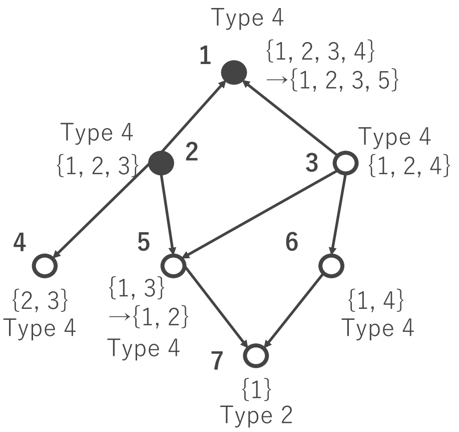
IDs 15 and 16 of Table I(b) describe the flux distributions when $\{g_1, g_2, g_4, g_5\}$ is deleted: $p_3 = g_1 = 0, p_4 = g_4 \vee g_5 = 0$ and $p_6 = g_2 \wedge g_3 = 0$ are obtained, v_3, v_4 and v_6 are forced to be 0, and no path is available. Thus, GR $=$ PR $= 0$ is obtained. Gene-deletion strategies $\{g_1, g_3, g_4, g_5\}$, $\{g_2, g_3, g_4, g_5\}$, $\{g_1, g_2, g_3, g_4, g_5\}$ are also classified as Type 8 as shown in Table I(a).

3) Behavior of RandTrimGdel: RandTrimGdel employs two sub-modules: GRPRchecker and DeletionShift. GRPRchecker returns the minimum PR at GR maximization when a constraint-based model and gene-deletion strategy are provided. DeletionShift equivalently converts a gene-deletion strategy so that the deletion of an alternative gene is replaced with that of a representative gene as much as possible.

In the network of Fig. 2, $\{g_1, g_2, g_4\}$ is a set of representative genes while $\{g_3, g_5\}$ is a set of alternative genes: g_3 is an alternative gene of a representative gene g_2; g_5 is an alternative gene of a representative gene g_4.

As shown in Table I(b), Types 2, 4, and 6 result in growth-coupled production, whereas the others do not. Because every strategy in Types 2, 4, or 6 includes $\{g_1\}$ or $\{g_2, g_3\}$, the minimal growth-coupling gene-deletion strategies are $\{g_1\}$ and $\{g_2, g_3\}$. Note that $\{g_2\}$ and $\{g_3\}$ are Type 3.

The minimal growth-coupling gene-deletion strategies for the example of Fig. 2 are obtained by RandTrimGdel as follows.



(a)

Target metabolite	m_5
Minimal gene-deletion strategies	$\{g_1\}, \{g_2, g_3\}$
Maximal gene-deletion strategies	$\{g_1, g_2, g_3, g_5\}$
→ Remaining genes	$\{g_4\}$
Replaceable genes	g_2: g_3
	g_4: g_5

(b)

Fig. 3. Minimal and maximal gene-deletion strategies are calculated using RandTrimGdel and AddGdel, starting with gDel_minRN strategies, respectively. Each node represents a growth-coupling gene-deletion strategy, where g_i is represented by i for visibility. (a) Nodes 1 and 2 represent the gDel_minRN strategies $\{g_1, g_2, g_3, g_5\}$ and $\{g_1, g_2, g_3\}$, respectively. RandTrimGdel derives the Nodes 4 and 7, $\{g_2, g_3\}$ and $\{g_1\}$, which are the minimal growth-coupling gene-deletion strategies. AddGdel derives Node 1 if Node 2 is given by gDel_minRN. (b) MetNetComp shows the minimal and maximal growth-coupling gene-deletion strategies and information on replaceable genes for each designated target metabolite and the constraint-based model.

First, RandTrimGdel employs gDel_minRN [24], which aims to find growth-coupling gene-deletion strategies that maximize the number of repressed reactions. Then, either $\{g_1, g_2, g_3\}$, $\{g_1, g_2, g_3, g_4\}$, or $\{g_1, g_2, g_3, g_5\}$ is obtained for D, by which r_2, r_3, and r_6 are repressed. If D contains an alternative gene g_j and does not have its representative gene g_i, then g_j is replaced with g_i. For example, if $D = \{g_1, g_2, g_3, g_5\}$ is obtained, then D is changed to $\{g_1, g_2, g_3, g_4\}$ because g_5 is an alternative gene of its representative gene g_4.

Suppose D obtained by gDel_minRN is $\{g_1, g_2, g_3\}$, which is represented by Node 2 in Fig. 3(a), where g_i is represented as i. The alternative gene is not replaced when D includes both the alternative gene and its representative gene. In this example, g_3 is not replaced with g_2 because D already includes g_2. Since $g_3 \in D^A$ holds, $D^R = \{g_1, g_2\}$.

Step 2 of RandTrimGdel iteratively and randomly selects a gene $g^r \in D^R$ and examines whether growth-coupled production is preserved when g^r is excluded from D. If preserved, g^r is excluded from D. If an alternative gene g^a of g^r is included in D, g^a is excluded from D and g^r is included in D again. Then, the random deletion of a gene is repeated for the updated D^R. If it is not preserved, the same process is conducted for another

$g \in D^R$. If not preserved for any $g \in D^R$, $D = D^R \cup D^A$ is saved as a minimal growth-coupling gene-deletion strategy.

For example, suppose that g_1 is selected from $D^R = \{g_1, g_2\}$ for Node 2 in Fig. 3(a). Then GRPRchecker is employed for $D = D^R \cup D^A = \{g_2\} \cup \{g_3\} = \{g_2, g_3\}$, which is represented by Node 4. Because Node 4 is Type 4, GRPRchecker returns success, and trimming g_1 from Node 2 is successful. Then, DeletionShift is applied to $D = D^R \cup D^A = \{g_2, g_3\}$, but D is not altered because the deletion of the alternative gene g_3 cannot be replaced with the deletion of the represented gene g_2.

If g_2 is selected from $D^R = \{g_1, g_2\}$ for Node 2 in Fig. 3(a), GRPRchecker is employed for $D = D^R \cup D^A = \{g_1\} \cup \{g_3\} = \{g_1, g_3\}$, which is represented by Node 5. Because $D = \{g_1, g_3\}$ is a Type 4 growth-coupling gene-deletion strategy, GRPRchecker returns success. Then, DeletionShift changes D from $\{g_1, g_3\}$ to $\{g_1, g_2\}$ because $g_2 \notin D^R$, $g_3 \in D^A$, and $Rep(g_3) = g_2$ hold. RandTrimGdel does not trim g_3 for Node 2 because g_3 is an alternative gene: $g_3 \in G^A$.

For Node 4 of Fig. 3(a), the candidate of trimming is only g_2 since g_3 is an alternative gene. However, $D = \{g_3\}$ is not a growth-coupling gene-deletion strategy as it is Type 3.

For Node 5, g_1 and g_2 are candidates for trimming because $g_1, g_2 \in G^R$. When g_1 is trimmed, $D = \{g_2\}$ is obtained, which is Type 3 and does not result in growth-coupling. When g_2 is trimmed, $D = \{g_1\}$ is obtained, which is a Type 2 growth-coupling gene-deletion strategy, represented by Node 7.

For Node 7, the trimming candidate is only g_1. When g_1 is deleted, $D = \phi$ is obtained, which is Type 1 and does not result in growth-coupling.

Thus, RandTrimGdel outputs $\mathbf{D} = \{\{g_1\}, \{g_2, g_3\}\}$. The MetNetComp database displays minimal growth-coupling gene-deletion strategies for m_5 as shown in Fig. 3(b). A list of replaceable genes is also shown. $g_2 : g_3$ and $g_4 : g_5$ mean $Rep(g_3) = g_2$ and $Rep(g_5) = g_4$, respectively.

4) Behavior of AddGdel: AddGdel employs three submodules: gDel_minRN, GRPRchecker, and RemainShift, which equivalently converts a gene-deletion strategy so that the deletion of a representative gene is replaced with its alternative gene as much as possible.

Similar to RandTrimGdel, AddGdel employs gDel_minRN in Step 1. Suppose that $D = \{g_1, g_2, g_3\}$ is obtained, which corresponds to Node 2 in Fig. 3(a).

In Step 2 of AddGdel, one of the remaining genes $G_{rem} = \{g_4, g_5\}$ is selected in ascending order of the index, and GRPRchecker is employed to check if growth-coupled production is maintained if the selected gene is deleted. First, g_4 is selected; GRPRchecker examines $\{g_1, g_2, g_3, g_4\}$ represented by Node 1 in Fig. 3(a), and returns success as it is Type 4 growth-coupling gene-deletion strategy. For Node 1, the remaining gene is only g_5: $G_{rem} = \{g_5\}$. Because $D = \{g_1, g_2, g_3, g_4, g_5\}$ is Type 8 and not a growth-coupling gene-deletion strategy, GRPRchecker returns false.

Subsequently, RemainShift is applied to $D = \{g_1, g_2, g_3, g_4\}$. As g_4 is the representative gene of an alternative gene g_5 and $(g_4, g_5) = (0, 1)$, RemainShift alters D to $\{g_1, g_2, g_3, g_5\}$, so that the remaining genes include as many representative genes as possible. Thus, AddGdel returns $\{g_4\}$ as the list of remaining genes for the maximal growth-coupling gene-deletion strategy, as shown in Fig. 3(b).

III. MetNetComp Database

A. Computational Experiments

The developed algorithms, RandTrimGdel and AddGdel, were applied to 11 constraint-based models, as shown in Table II. All procedures in the computational experiments were implemented using CentOS 7 machine with an AMD Ryzen Processor with 2.90 GHz 64 cores/128 threads, 128 GB memory, and 1 TB SSD. This workstation had a CPLEX 12.10, COBRA Toolbox v3.0 [28], and MATLAB R2021a installed and used for these analyses. An auxiliary exchange reaction was temporarily added to the model to simulate target metabolite production if the target metabolite did not have a production reaction.

Table II presents the number of genes, metabolites, reactions, the target metabolites for which the minimal and maximal gene-deletion strategies were obtained, and the obtained minimal and maximal growth-coupling gene-deletion strategies. For e_coli_core, iLJ478, and iNF517, another growth-coupling gene-deletion strategy was not obtained for the recent 100 loops of RandTrimGdel. This implies that the number of growth-coupling gene deletion strategies was almost saturated for these three models. For the other models, however, it was not saturated. Table III represents the average computation time for computing a gene-deletion strategy for a target metabolite using gDel_minRN, Step 2 of RandTrimGdel, and Step 2 of AddGdel, where gDel_minRN is Step 1 for both gDel_minRN and RandTrimGdel.

B. Top Page and Index

Fig. 4 shows a screenshot of the top page of the MetNetComp database. Users can select their target metabolite using one of the following three methods: (1) Select the target metabolite in a constraint-based model. (2) Select a target metabolite using the initial letter. (3) Directly search for a target metabolite.

Suppose the target metabolite is succinate, succ_e. Users can select succ_e by selecting the initial letter S or by directly searching for succ_e. Then, the list of constraint-based models for which growth-coupling gene-deletion strategies for succ_e are available: e_coli_core, iML1515, and iMM904 are applicable in this case.

C. Main Page

Suppose that the target metabolite and constraint-based model are designated as succ_e and e_coli_core, respectively. The main page includes the following information.

There were 47 minimal growth-coupling gene-deletion strategies for succ_e in e_coli_core, excluding apparent duplicate counting by replaceable genes. The 47 strategies were sorted in ascending order of the number of gene deletions. For each strategy, the genes to be deleted are shown and linked to the corresponding pages in the KEGG database, excluding iNF517 and iHN637 because KEGG does not have corresponding entries.

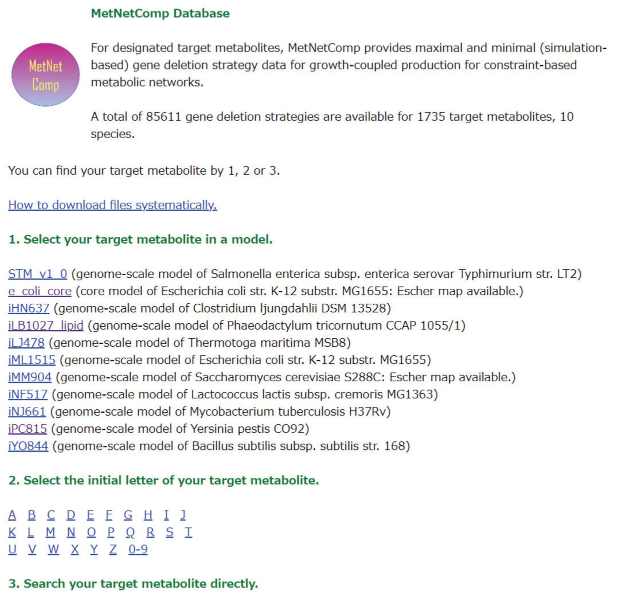
TABLE II
THE NUMBER OF GENES, METABOLITES, AND REACTIONS FOR EACH CONSTRAINT-BASED MODEL, FOR WHICH RANDTRIMGDEL AND ADDGDEL WERE APPLIED

Model	Species	#genes	#mets	#reactions	#targets	#strategies	
e_coli_core	*Escherichia coli*	137	72	95	41	1020	(almost saturated)
iLJ478	*Thermotoga maritima*	482	570	652	98	521	(almost saturated)
iNF517	*Lactococcus lactis*	516	650	754	3	214	(almost saturated)
iHN637	*Clostridium ljungdahlii*	637	698	785	202	3785	(almost saturated)
iNJ661	*Mycobacterium tuberculosis*	661	825	1025	118	6758	
iPC815	*Yersinia pestis*	815	1552	1961	85	1679	
iYO844	*Yersinia pestis*	844	990	1250	9	522	
iMM904	*Saccharomyces cerevisiae*	905	1226	1577	105	2854	
iLB1027_lipid	*Phaeodactylum*	1027	2172	4456	1095	29981	
STM_v1_0	*Salmonella enterica*	1271	1802	2545	161	11022	
iML1515	*Escherichia coli*	1516	1877	2712	442	27255	

The models are sorted in ascending order of the number of genes. The number of target metabolites and gene-deletion strategies, derived by RandTrimGdel and addgdel, available in MetNetComp database is also shown.

TABLE III
AVERAGE COMPUTATION TIME FOR A TARGET METABOLITE OF EACH CONSTRAINT-BASED MODEL IN STEP 1 OF RANDTRIMGDEL AND ADDGDEL, STEP 2 OF RANDTRIMGDEL, AND STEP 2 OF ADDGDEL

Model	Step 1 (gDel_minRN)	Step 2 of RandTrimGdel	Step 2 of AddGdel
e_coli_core	0.12s	9.39s	0.15s
iLJ478	3m24s	47.6s	1m44s
iNF517	1.6s	1m11s	13.6s
iHN637	1.3s	1m38s	1m52s
iNJ661	6.7s	4m2s	4m27s
iPC815	43.7s	11m47s	12m13
iYO844	23.5s	9m34s	4m31s
iMM904	49.2s	26m41s	7m25s
iLB1027_lipid	29.7s	22m42s	18m37s
STM_v1_0	51.8s	50m57s	20m19s
iML1515	7m35s	1h38m32s	20m8s


MetNetComp Database

For designated target metabolites, MetNetComp provides maximal and minimal (simulation-based) gene deletion strategy data for growth-coupled production for constraint-based metabolic networks.

A total of 85611 gene deletion strategies are available for 1735 target metabolites, 10 species.

You can find your target metabolite by 1, 2 or 3.

How to download files systematically.

1. Select your target metabolite in a model.

STM_v1_0 (genome-scale model of Salmonella enterica subsp. enterica serovar Typhimurium str. LT2)
e_coli_core (core model of Escherichia coli str. K-12 substr. MG1655: Escher map available.)
iHN637 (genome-scale model of Clostridium ljungdahlii DSM 13528)
iLB1027_lipid (genome-scale model of Phaeodactylum tricornutum CCAP 1055/1)
iLJ478 (genome-scale model of Thermotoga maritima MSB8)
iML1515 (genome-scale model of Escherichia coli str. K-12 substr. MG1655)
iMM904 (genome-scale model of Saccharomyces cerevisiae S288C: Escher map available.)
iNF517 (genome-scale model of Lactococcus lactis subsp. cremoris MG1363)
iNJ661 (genome-scale model of Mycobacterium tuberculosis H37Rv)
iPC815 (genome-scale model of Yersinia pestis CO92)
iYO844 (genome-scale model of Bacillus subtilis subsp. subtilis str. 168)

2. Select the initial letter of your target metabolite.

A B C D E F G H I J
K L M N O P Q R S T
U V W X Y Z 0-9

3. Search your target metabolite directly.

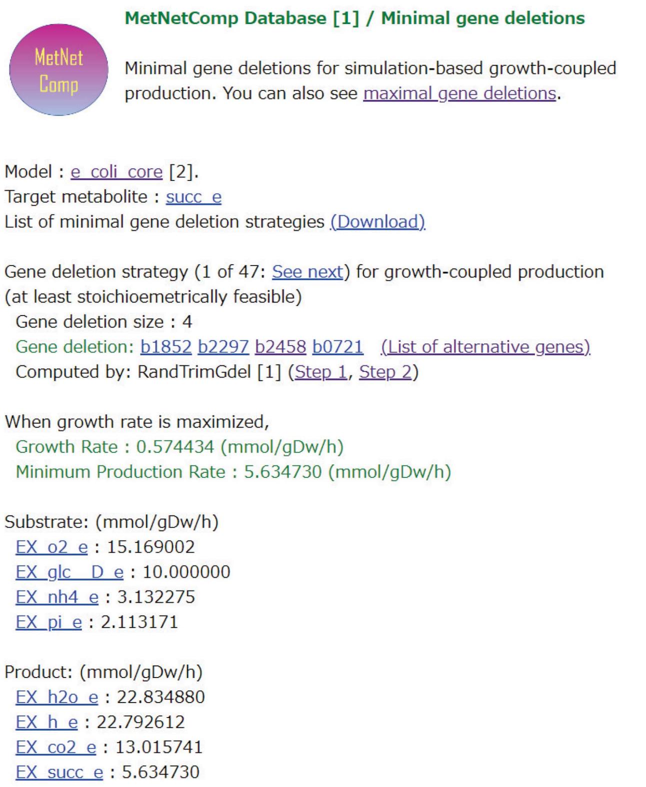
Fig. 4. Screenshot of the top page of the MetNetComp database. Users can select the target metabolites (1) in a selected constraint-based model, (2) using an initial letter, or (3) by direct search. 85,611 gene-deletion strategies for 1,735 target metabolites, 10 species, and 11 constraint-based models are available.

Fig. 5 is the screenshot of the main web page for the first gene-deletion strategy, which includes four gene deletions: b1852, b2297, b2458, and b0721. The list of alternative genes is shown as ”b0721: b0722 b0724 b0723,” which indicates that b0721 is the representative gene of b0722, b0724, and b0723. The GR and the minimum PR at GR maximization are also shown in this example: 0.574 and 5.63 (mmol/gDw/h). The reaction rates of the substrates were

EX_o2_e: 15.17,
EX_glc__D_e: 10,
EX_nh4_e: 3.13,
EX_pi_e: 2.11 (mmol/gDw/h),

which represent the uptake reaction rates of oxygen, glucose, ammonia, and phosphate, respectively. The reaction rates of the products were

EX_h2o_e: 22.83,
EX_h_e: 22.79,
EX_co2_e: 13.01,


MetNetComp Database [1] / Minimal gene deletions

Minimal gene deletions for simulation-based growth-coupled production. You can also see maximal gene deletions.

Model : e_coli_core [2].
Target metabolite : succ_e
List of minimal gene deletion strategies (Download)

Gene deletion strategy (1 of 47: See next) for growth-coupled production (at least stoichioemetrically feasible)
Gene deletion size : 4
Gene deletion: b1852 b2297 b2458 b0721 (List of alternative genes)
Computed by: RandTrimGdel [1] (Step 1, Step 2)

When growth rate is maximized,
Growth Rate : 0.574434 (mmol/gDw/h)
Minimum Production Rate : 5.634730 (mmol/gDw/h)

Substrate: (mmol/gDw/h)
EX_o2_e : 15.169002
EX_glc__D_e : 10.000000
EX_nh4_e : 3.132275
EX_pi_e : 2.113171

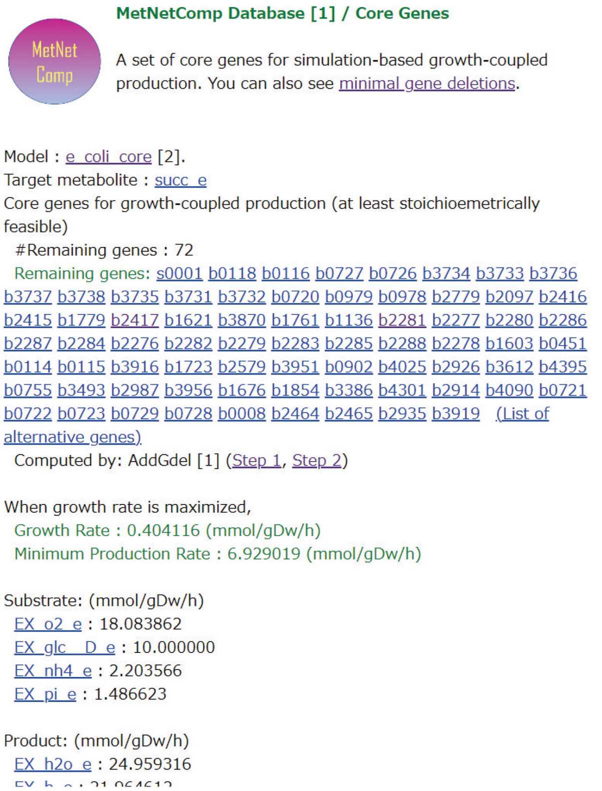
Product: (mmol/gDw/h)
EX_h2o_e : 22.834880
EX_h_e : 22.792612
EX_co2_e : 13.015741
EX_succ_e : 5.634730

Fig. 5. Screenshot of the MetNetComp webpage representing a minimal growth-coupling gene-deletion strategy for succinate in e_coli_core.

TABLE IV
CSV Files for Minimal and Maximal Gene Deletion Strategies and Equivalent Genes, as Well as JSON Files for Reaction Rates Related to These Strategies, Can Be Systematically Downloaded Using the Provided URL

Type		URL
1	Gene deletion strategies	https://metnetcomp.github.io/database1/csv/<model>&<target metabolite>.csv
2		https://metnetcomp.github.io/database1/csv/<model>&<target metabolite>&core.csv
3	Equivalent genes	https://metnetcomp.github.io/database1/csv/<model>&equivalentGenes.csv
4	Reaction rates	https://metnetcomp.github.io/database1/json/<model>&<target metabolite>&minimal&<id>.json
5		https://metnetcomp.github.io/database1/json/<model>&<target metabolite>&core.json

How to form the URL for the systematic download is illustrated with examples at https://metnetcomp.Github.Io/database1/indexfiles/systematicdownload.Html.



MetNetComp Database [1] / Core Genes

A set of core genes for simulation-based growth-coupled production. You can also see minimal gene deletions.

Model : e_coli_core [2].
Target metabolite : succ_e
Core genes for growth-coupled production (at least stoichioemetrically feasible)
#Remaining genes : 72
Remaining genes: s0001 b0118 b0116 b0727 b0726 b3734 b3733 b3736 b3737 b3738 b3735 b3731 b3732 b0720 b0979 b0978 b2779 b2097 b2416 b2415 b1779 b2417 b1621 b3870 b1761 b1136 b2281 b2277 b2280 b2286 b2287 b2284 b2276 b2282 b2279 b2283 b2285 b2288 b2278 b1603 b0451 b0114 b0115 b3916 b1723 b2579 b3951 b0902 b4025 b2926 b3612 b4395 b0755 b3493 b2987 b3956 b1676 b1854 b3386 b4301 b2914 b4090 b0721 b0722 b0723 b0729 b0728 b0008 b2464 b2465 b2935 b3919 (List of alternative genes)
Computed by: AddGdel [1] (Step 1, Step 2)

When growth rate is maximized,
Growth Rate : 0.404116 (mmol/gDw/h)
Minimum Production Rate : 6.929019 (mmol/gDw/h)

Substrate: (mmol/gDw/h)
EX_o2_e : 18.083862
EX_glc__D_e : 10.000000
EX_nh4_e : 2.203566
EX_pi_e : 1.486623

Product: (mmol/gDw/h)
EX_h2o_e : 24.959316
EX_h_e : 21.964613

Fig. 6. Screenshot of the MetNetComp webpage representing a maximal growth-coupling gene-deletion strategy for succinate in e_coli_core.

EX_succ_e: 5.63 (mmol/gDw/h),
which represent the production reaction rates of water, hydrogen ion, carbon dioxide, and succinate, respectively.

The maximal gene-deletion strategy is also shown: the number of remaining genes is 72, and the number of deleted genes is 137 −72 = 65. Fig. 6 is the screenshot of the web page of the maximal gene-deletion strategy for succ_e and e_coli_core. The web pages of the minimal and maximal gene-deletion strategies are linked to each other.

D. Visualization

For each gene-deletion strategy, the resulting flux distribution is described by a JSON file and can be downloaded. Among 11 constraint-based models, for ecolicore, iML1515, and iMM904, visualization by Escher [26] is possible using the downloaded JSON files, where the flux distribution data for iML1515 is converted to that for iJO1366. Fig. 7 represents the visualization of flux distribution for the gene-deletion strategy for succ_e and e_coli_core.

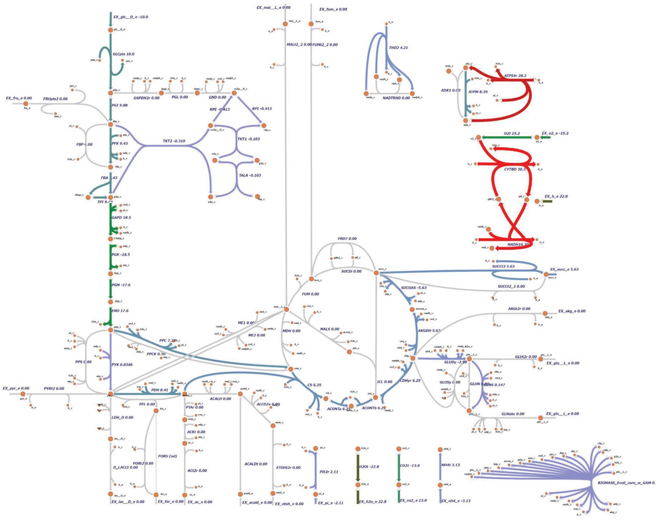


Fig. 7. Visualization by Escher [26] for a minimal gene-deletion strategy for succinate in e_coli_core.

E. Download

The CSV and JSON files can be systematically downloaded using the URLs provided in Table IV. CSV files for minimal and maximal gene deletion strategies can be downloaded via the URLs with Types 1 and 2 in Table IV after specifying a model and a target metabolite. A CSV file for equivalent genes can be accessed via the URL with Type 3 in Table IV after specifying a model. JSON files for reaction rates can be downloaded via the URLs with Types 4 and 5 in Table IV after specifying a model and a target metabolite. The <id> is a positive integer that represents the order when the gene deletion strategies are sorted by the number of deletions.

IV. Discussion

A. MetNetComp Database

MetNetComp provides 85,611 growth-coupling gene-deletion strategies for 1,735 target metabolites, 10 species, and 11 constraint-based models. This information is available to users even without the knowledge of computational methods or resources.

The Bigg Models database contains 108 constraint-based models, whereas MetNetComp contains gene-deletion strategies for only 11 constraint-based models. If Bigg Models database contains multiple constraint-based models for one species, one or two of the most widely used models were selected, and computational experiments were conducted for MetNetComp. For example, the Bigg Models database contains 58 constraint-based models for *Escherichia coli*, and MetNetComp contains only e_coli_core and iML1515. If no growth-coupling gene-deletion strategies were obtained for any target metabolite, MetNetComp did not contain gene-deletion strategies for the models.

Because all the gene-deletion strategies in MetNetComp were obtained by modifying the gDel_minRN strategies, improving the performance of gDel_minRN is a future task. Analyzing gene-deletion strategies for different culture conditions, such as anaerobic conditions and LB medium, and for models that are not present in the Bigg Models database is also required.

When a target metabolite and constraint-based model are designated, a maximal gene-deletion strategy and many minimal gene-deletion strategies are listed. If a pair of minimal and maximal gene-deletion strategies with k_1 and k_2 gene deletions is designated, there exists at least $k_2 - k_1 - 1$ and at most $2^{k_2-k_1-1}$ growth-coupling gene-deletion strategies between them. Theoretical and experimental analyses of the extent to which $2^{k_2-k_1-1}$ gene-deletion strategies result in growth-coupling are also a required task in the future.

MetNetComp lists only one maximal gene-deletion strategy whereas it lists many minimal gene-deletion strategies for the following reasons: (1) Because gDel_minRN maximizes the number of repressed reactions, the obtained gene-deletion strategy is close to the maximum number of gene deletions; (2) In metabolic engineering, the enumeration of many candidates is required for minimal gene-deletion strategies but not for maximal gene-deletion strategies. Maximal gene-deletion strategies help analyze and understand the core networks for growth-coupled production from a biological perspective.

B. Randtrimgdel and Addgdel

To compute minimal gene-deletion strategies, RandTrimGdel trims only the representative genes. Because every alternative gene has its representative gene and they are replaceable, it is sufficient to consider representative genes for trimming. For example, suppose that g^a is an alternative gene of its representative gene g^r, and both are included in a growth-coupling gene-deletion strategy. It is clear that trimming g^a is equivalent to trimming g^r. However, trimming both g^a and g^r may lead to another growth-coupling gene-deletion strategy. Therefore, when g^r is trimmed, RandTrimGdel employs DeletionShift to shift the deletion from g^a to g^r: $(g^r, g^a) = (1, 0)$ is changed to $(g^r, g^a) = (0, 1)$. This allows RandTrimGdel to not exclude any minimal growth-coupling gene-deletion strategies for exploration while limiting trimming to representative genes.

RandTrimGdel does not guarantee that it will exhaustively find all minimal growth-coupling gene-deletion strategies as it is a randomized algorithm. As shown in Table II, almost all minimal growth-coupling gene-deletion strategies could be enumerated by RandTrimGdel for the constraint-based models with less than 650 genes. Ideally, we should develop a deterministic algorithm that exhaustively finds every minimal growth-coupling gene-deletion strategy but is expected to be difficult to apply to genome-scale models because of the combinatorial explosion in the number of candidate solutions.

To compute a maximal gene-deletion strategy, AddGdel starts with the gDel_minRN strategy and adds gene deletions. AddGdel does not distinguish between representative and alternative genes when increasing the number of gene deletions. However, in the last process, the deletion of representative genes is replaced by the deletion of alternative genes as much as possible. Thus, the MetNetComp database represents the genes necessary for growth-coupled production using representative genes as far as possible.

As shown in Tables II and III, for constraint-based models with more than 900 genes, more than 20 minutes were required to compute a minimal growth coupling gene-deletion strategy using RandTrimGdel. Users of the MetNetComp database can save computation time and resources, particularly for large models. The MetNetComp database provides a large number of gene-deletion strategies. This could lead to a new research paradigm that uses machine learning to analyze and design constraint-based metabolic networks.

C. Time and Space Complexity Analysis

Both RandTrimGdel and AddGdel solve mixed integer linear programming (MILP) once in Step 1 and then solve LP repeatedly in Step 2. Because MILP is an NP-complete problem [25], the computation time is proportional to an exponential function of the problem size. Because the number of linear programming iterations is $O(g^2)$, and $O(g)$ time is required for DeletionShift and RemainShift, the computation time of Step 2 is proportional to a polynomial function of the problem size. However, the actual computation time of Step 2 of RandTrimGdel and AddGdel was longer than Step 1 as shown in Table III.

The computation time of Step 2 of RandTrimGdel is almost in ascending order of the number of genes: only the order for pairs (iPC815, iYO844) and (iMM904, iLB1027_lipid) is different. This tendency is the same for Step 2 of AddGdel, although it is weakened slightly and drastically for Step 1 of RandTrimGdel and AddGdel. The computation time of Step 2 is often significantly longer than Step 1, which solves NP-complete problems, even though Step 2 employs polynomial-time algorithms. This is because more computation time is required to guarantee the minimality and maximality of the solutions in Step 2, whereas Step 1 does not ensure either the minimality or maximality.

To hold S minimal gene deletion strategies for a target metabolite, $O(S \cdot |D|)$ space is required when $|D|$ represents the average gene deletion size. If we do not focus on representative genes in the enumeration, $O(S \cdot |D| \cdot \Pi_{i=1}^{|D|} \max(|alt(g_i^R)|, 1))$ space is required where $|alt(g_i^R)|$ represents the number of alternative genes for a representative deleted gene g_i^R.

References

[1] N. Hillson et al., “Building a global alliance of biofoundries,” *Nature Commun.*, vol. 10, no. 1, pp. 1–4, 2019.

[2] J. D. Orth, I. Thiele, and B. O. Palsson, "What is flux balance analysis?," *Nature Biotechnol.*, vol. 28, no. 3, pp. 245–248, 2010.

[3] W. B. Copeland et al., "Computational tools for metabolic engineering," *Metabolic Eng.*, vol. 14, no. 3, pp. 270–280, 2012.

[4] A. P. Burgard, P. Pharkya, and C. D. Maranas, "Optknock: A bilevel programming framework for identifying gene knockout strategies for microbial strain optimization," *Biotechnol. Bioeng.*, vol. 84, no. 6, pp. 647–657, 2003.

[5] P. Pharkya, A. P. Burgard, and C.D. Maranas, "OptStrain: A computational framework for redesign of microbial production systems," *Genome Res.*, vol. 14, no. 11, pp. 2367–2376, 2004.

[6] P. Pharkya and C. D. Maranas, "An optimization framework for identifying reaction activation/inhibition or elimination candidates for overproduction in microbial systems," *Metabolic Eng.*, vol. 8, no. 1, pp. 1–13, 2006.

[7] P. R. Patil, I. Rocha, J. Förster, and J. Nielsen, "Evolutionary programming as a platform for in silico metabolic engineering," *BMC Bioinf.*, vol. 6, no. 1, 2005, Art. no. 308.

[8] S. Ranganathan, P. F. Suthers, and C. D. Maranas, "OptForce: An optimization procedure for identifying all genetic manipulations leading to targeted overproductions," *PLoS Comput Biol*, vol. 6, no. 4, 2010, Art. no. e1000744.

[9] I. Rocha et al., "OptFlux: An open-source software platform for in silico metabolic engineering," *BMC Syst. Biol.*, vol. 4, no. 1, pp. 1–12, 2010.

[10] Y. Toya and H. Shimizu, "Flux analysis and metabolomics for systematic metabolic engineering of microorganisms," *Biotechnol. Adv.*, vol. 31, no. 6, pp. 818–826, 2013.

[11] D. S. Lun et al., "Large-scale identification of genetic design strategies using local search," *Mol. Syst. Biol.*, vol. 5, no. 1, 2009, Art. no. 296.

[12] G. Rockwell, N. J. Guido, and G. M. Church, "Redirector: Designing cell factories by reconstructing the metabolic objective," *PLoS Comput Biol*, vol. 9, no. 1, 2013, Art. no. e1002882.

[13] L. Yang, W. R. Cluett, and R. Mahadevan, "EMILiO: A fast algorithm for genome-scale strain design," *Metabolic Eng.*, vol. 13, no. 3, pp. 272–281, 2011.

[14] D. Egen and D. S. Lun, "Truncated branch and bound achieves efficient constraint-based genetic design," *Bioinformatics*, vol. 28, no. 12, pp. 1619–1623, 2012.

[15] N. E. Lewis et al., "Omic data from evolved *E. coli* are consistent with computed optimal growth from genome-scale models," *Mol. Syst. Biol.*, vol. 6, no. 1, 2010, Art. no. 390.

[16] D. Gu, C. Zhang, S. Zhou, L. Wei, and Q. Hua, "IdealKnock: A framework for efficiently identifying knockout strategies leading to targeted overproduction," *Comput. Biol. Chem.*, vol. 61, pp. 229–237, 2016.

[17] S. Ohno, H. Shimizu, and C. Furusawa, "FastPros: Screening of reaction knockout strategies for metabolic engineering," *Bioinformatics*, vol. 30, no. 7, pp. 981–987, 2014.

[18] T. Tamura, "Grid-based computational methods for the design of constraint-based parsimonious chemical reaction networks to simulate metabolite production: GridProd," *BMC Bioinf.*, vol. 19, no. 1, 2018, Art. no. 325.

[19] P. Schneider, P. S. Bekiaris, A. von Kamp, and S. Klamt, "StrainDesign: A comprehensive Python package for computational design of metabolic networks," *Bioinformatics*, vol. 38, no. 21, pp. 4981–4983, 2022.

[20] D. Machado, M. J. Herrgård, and I. Rocha, "Stoichiometric representation of gene–protein–reaction associations leverages constraint-based analysis from reaction to gene-level phenotype prediction," *PLoS Comput. Biol.*, vol. 12, no. 10, 2016, pp. e1005140.

[21] Z. Razaghi-Moghadam and Z. Nikoloski, "GeneReg: A constraint-based approach for design of feasible metabolic engineering strategies at the gene level," *Bioinformatics*, vol. 37, no. 12, pp. 1717–1723, 2020.

[22] T. Tamura, "Trimming gene deletion strategies for growth-coupled production in constraint-based metabolic networks: TrimGdel," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 20, no. 2, pp. 1540–1549, 2023.

[23] C. J. Norsigian, "BiGG models 2020: Multi-strain genome-scale models and expansion across the phylogenetic tree," *Nucleic Acids Res.*, vol. 48, no. D1, pp. D402–D406, 2020.

[24] T. Tamura, A. Muto-Fujita, Y. Tohsato, and T. Kosaka, "Gene deletion algorithms for minimum reaction network design by mixed-integer linear programming for metabolite production in constraint-based models: GDel_minRN," *J. Comput. Biol.*, vol. 30, no. 5, pp. 553–568, 2023.

[25] D. Gusfield, *Integer Linear Programming in Computational and Systems Biology: An Entry-Level Text and Course*. Cambridge, U.K.: Cambridge Univ. Press, 2019.

[26] Z. A. King, A. Dräger, A. Ebrahim, N. Sonnenschein, N. E. Lewis, and B. O. Palsson, "Escher: A web application for building, sharing, and embedding data-rich visualizations of biological pathways," *PLoS Comput Biol*, vol. 11, no. 8, 2015, Art. no. e1004321.

[27] M. Kanehisa and S. Goto, "KEGG: Kyoto Encyclopedia of genes and genomes," *Nucleic Acids Res.*, vol. 28, no. 1, pp. 27–30, 2000.

[28] L. Heirendt et al., "Creation and analysis of biochemical constraint-based models using the COBRA toolbox v. 3.0," *Nature Protoc.*, vol. 14, no. 3, pp. 639–702, 2019.

Takeyuki Tamura (Member, IEEE) received the BE, ME, and PhD degrees in informatics from Kyoto University, Japan, in 2001, 2003, and 2006, respectively. He joined Bioinformatics Center, Institute for Chemical Research, Kyoto University as a postdoctoral fellow in April 2006. He worked as an assistant professor from December 2007 to September 2017 and started working as an associate professor from October 2017. His research interests include mathematical metabolic engineering based on algorithm theory.