

# Elementary flux mode 型代謝経路の化学量論的に釣り合った代謝産物レベルネットワーク構造算出の高速化

太田潤<sup>†1,a)</sup>

**概要** : 代謝ネットワークにおける“経路”として、基質-生成物の二項関係を連ねたタイプの“経路”以外に、化学量論的に釣り合った個々の反応の集合としての“経路”を考慮することができる。これまで、反応の集合としての“経路”を elementary flux mode 型経路 (EFM 型経路) と呼び、通常は原子レベルで考えることのない EFM 型経路に原子レベルの情報を与える (原子レベルマッピングを行う) ためのアルゴリズムを報告してきた。EFM 型経路の原子レベルマッピングでは、副産物として、EFM 型経路に含まれる反応が原料分子群から目的分子群までどのような順序で働くかを示す、経路の化学量論的に釣り合った代謝産物レベルネットワーク構造も算出される。前稿において、原子レベルの情報を含まない代謝産物レベル反応情報のみから、EFM 型経路の化学量論的に釣り合った代謝産物レベルネットワーク構造を算出するための基本アルゴリズムを、異同判定アルゴリズム (基本アルゴリズムによりネットワーク構造が複数算出された場合にそれらの異同判定を行う) とともに報告した。本稿では、大幅に高速化された、EFM 型経路の化学量論的に釣り合った代謝産物レベルネットワーク構造算出アルゴリズムを報告する。報告するアルゴリズムでは、基本アルゴリズムにおける置換を互換の繰り返しとして行い、互換によりネットワーク構造が算出されるごとにネットワーク構造の異同判定を行う。

**キーワード** : 代謝ネットワーク, EMU tracing, elementary flux mode

## Accelerating calculation of stoichiometrically balanced metabolite-level network structure of elementary flux mode type metabolic pathway

JUN OHTA<sup>†1,a)</sup>

### 1. はじめに

生理活性物質の生成は、代謝ネットワークの主要な機能であり、生理活性物質の生成経路に関する知識・情報は、医学・生物学的に重要である。これまで、代謝ネットワークにおける経路 (代謝経路) として研究されてきたものには 2 種類の考え方の“経路”, 原料代謝産物に含まれる原子が基質 - 生成物の二項関係の連なりにより目的代謝産物に含まれる原子に移動する道筋としての“経路” [1-6] と化学量論的に釣り合った個々の反応の集合としての“経路” [7,8] が含まれる。化学量論的に釣り合った個々の反応の集合としての“経路”の代表例は, elementary flux mode [7] と extreme pathway [8] であり, 本稿ではこれらの例を含む後者の“経路”を elementary flux mode 型経路 (EFM 型経路) と呼ぶ。

特定の目的代謝産物 B の生成原料として代謝産物 A が適しているか判断する際に, 通常, A に含まれる原子を B に含まれる原子に移動させる linear な経路 (基質 - 生成物の二項関係の連なり) の中で, A の分子を構成する原子のうちより多くをより少ない数の反応で B まで移動させるものが代謝マップ上で検討されるが, A から B に至る linear な経路の存在は, A が B の原料であるための必要条件であ

り, A のみを原料として B が生成することを保証しない。一方, 反応の集合である EFM 型経路においては, 反応の集合から求まる経路の収支式左辺に現れる原料分子群のすべてが揃った場合に収支式右辺の目的分子群が生成すると言えるが, 反応の集合そのものは, 原料分子群から目的分子群に至る具体的な道筋 (EFM 型経路に含まれる反応がどのような順序で働くか) を明示しない。

通常は原子レベルで考えることのない EFM 型経路について, 私は, “EFM 型経路における原料分子群と目的分子群は化学量論的に釣り合っており, 基本的には 1 セットの原料分子群の各原子が 1 セットの目的分子群の各原子に取り込まれる原子レベルの道筋が存在する筈である”と考え, EFM 型経路に原子レベルの情報を与える (原子レベルマッピングを行う) ためのアルゴリズムを研究してきた [9,10,11]。その過程で, EFM 型経路の原子レベルマッピングの副産物として経路の代謝産物レベルの代謝ネットワーク構造が求まることを経験した。一般的な代謝マップが, 化学量論を考慮していない (解糖系では化学量論的に 1 分子の glucose から 2 分子の lactate が生成するが代謝マップ上には glucose ノードと lactate ノードがそれぞれ 1 つしかない等) ことが多いのに対して, EFM 型経路の原子レベルマッピングの副産物として求まる代謝産物レベルの代謝ネットワーク構造は, 経路の化学量論を正確に表現する, 化学量論的に釣り合った代謝ネットワーク構造であった。ま

<sup>†1</sup> 岡山大学 大学院医歯薬学総合研究科 (医) 生化学分野  
a) jo25@md.okayama-u.ac.jp

た, 1つのEFM型経路に対して可能な化学量論的に釣り合った代謝産物レベルネットワーク構造は1種類とは限らなかった. ペントースリン酸経路の反応により2分子のfructose 6-phosphateと1分子のglyceraldehyde 3-phosphateから3分子のribose 5-phosphateが生成するEFM型経路の完全原子レベルマッピングにおいては, 1つのEFM型経路に対して2種類の化学量論的に釣り合った代謝産物レベルネットワーク構造(cycleを含まない経路とcycleを含む経路)が算出された[9,10],

EFM型経路の化学量論的に釣り合った代謝産物レベルネットワーク構造は, 原料分子群から目的分子群に至る具体的な道筋(EFM型経路に含まれる反応がどのような順序で働くか)を示す, EFM型経路の理解に不可欠なものである. これまでの代謝産物レベルの生合成代謝経路に関する情報学的研究には, 目的代謝産物の生成に必要な原料代謝産物の集合を求める手法の研究[12]や与えられた代謝ネットワークに存在する生合成経路の列挙を試みる研究[13]などがある. 前者の手法では生合成経路の算出はできない. 後者の手法では生合成経路が産出されるが, 算出されるのは, 化学量論を考慮しない1代謝産物(種)1ノードの代謝産物レベルネットワーク構造である.

1つのEFM型経路に, EFM型経路の化学量論的に釣り合った代謝産物レベルネットワーク構造が複数あれば, それぞれのネットワーク構造に対応する原子レベルマッピングが存在すると考えられるので, EFM型経路を構成する反応の原子レベル情報を用いて, EFM型経路の化学量論的に釣り合った代謝産物レベルネットワーク構造をEFM型経路の原子レベルマッピングにより求めることには一定の合理性がある. しかし, 原理的に, 代謝産物レベルのネットワーク構造は原子レベル情報に依存しない. そこで, 原子レベルの情報を用いることなく, EFM型経路を構成する反応の代謝産物レベルの化学量論情報のみからEFM型経路の化学量論的に釣り合った代謝産物レベルネットワーク構造を算出する方法の研究を行い, 前稿[14]では, EFM型経路の化学量論的に釣り合った代謝産物レベルネットワーク構造を求めるための基本アルゴリズム(基本アルゴリズム)とEFM型経路の化学量論的に釣り合った代謝産物レベルネットワーク構造の異同判定アルゴリズム(異同判定アルゴリズム)およびそれらを用いた計算実験の結果を述べた. 異同判定アルゴリズムは, 基本アルゴリズムによりネットワーク構造が複数算出された場合に, それらの異同判定を行うためのものである.

本稿では, 前稿[14]で報告した“基本アルゴリズムでネットワーク構造を算出した後に, ネットワーク構造の異同判定を行って, uniqueなネットワーク構造を求める”手順を, アルゴリズムAと呼ぶ. アルゴリズムAでは, 同じ代謝産物が多数回現れる, あるいは, 複数回現れる代謝産物が多いEFM型経路の場合, 算出される構造が多いため,

基本アルゴリズムのStep 10(置換を含む)と異同判定に要する計算時間が膨大となる. 本稿では, 基本アルゴリズムにおける置換を互換の繰り返しとして行い, 互換によりネットワーク構造が算出されるごとにネットワーク構造の異同判定を行う, 大幅に高速化された, EFM型経路の化学量論的に釣り合った代謝産物レベルネットワーク構造算出アルゴリズム(アルゴリズムB)を報告する.

## 2. 方法

アルゴリズムBを, 前稿[14]で報告した基本アルゴリズムと異同判定アルゴリズムから構成されるアルゴリズムAの概要とともに説明する. また, アルゴリズムA, Bを高速化するための付加ステップも説明する. アルゴリズムA, Bにより直接に算出される“化学量論的に釣り合った代謝産物レベルネットワーク構造”は, 2.1に述べる変換後のEFM型経路の, 代謝産物と反応をノードとするネットワーク構造であり, すべての代謝産物ノードが, そのノードが示す代謝産物分子を生成する反応の反応ノードと利用する反応の反応ノードの計2ノードのみとつながりを持つ. 本稿では, 代謝産物に対応するノードを“代謝産物ノード”, 反応に対応するノードを“反応ノード”と呼ぶ. 以下において“化学量論的な代謝ネットワーク構造”は“化学量論的に釣り合った代謝産物レベルネットワーク構造”と同義である.

### 2.1 与えられたEFM型経路の, 単一の仮想原料反応から単一の仮想目的反応に至るEFM型経路への変換

アルゴリズムAとアルゴリズムBの何れにおいても, 構成反応の集合として与えられたEFM型経路から, 経路の原料代謝産物source(s)と目的代謝産物target(s)の情報を得る. 原料代謝産物と目的代謝産物は何れも複数個の分子である場合がある. この原料代謝産物と目的代謝産物に対して, 原料代謝産物を何も無いところから生成する仮想原料反応source reactionと目的代謝産物を除去するが何も生成しない仮想目的反応target reactionを定義して反応の集合に追加し, 与えられたEFM型経路を単一の仮想原料反応source reactionから単一の仮想目的反応target reactionへの経路に変換する.

source reaction: no metabolite(s) → source(s)

target reaction: target(s) → no metabolite(s)

この変換により, 原料代謝産物と目的代謝産物の代謝産物ノードを含むすべての代謝産物ノードが例外なくそのノードが示す分子を生成する反応の反応ノードと利用する反応の反応ノードの計2ノードのみとつながりを持つEFM型経路のネットワーク構造を考えられるようになる. 2.2から2.6における“EFM型経路”は変換後のEFM型経路を指す.

## 2.2 EFM 型経路の反応構成の反応番号による記述および反応番号と代謝産物番号による反応の記述

アルゴリズム A とアルゴリズム B の何れにおいても、EFM 型経路に現れる反応、代謝産物に対して、反応番号、代謝産物番号を付す。ある反応・代謝産物が経路（の同じコンパートメント）に複数回現れる場合、それらを同種とみなし、それらに同じ反応番号・代謝産物番号を付す。

EFM 型経路の反応構成を反応番号の配列（反応番号配列）として表現する。EFM 型経路に反応番号  $r$  の反応が  $n$  回現れる場合は、反応番号配列に  $r$  が  $n$  回現れるようにする。各反応の反応番号を反応番号配列に格納する順序は任意であるが、いったん格納したら番号の順序を変更しない。

EFM 型経路に現れる反応はそれぞれ、反応番号、基質の代謝産物番号の配列、生成物の代謝産物番号の配列の組み合わせとして表現する。ある反応の反応式の基質側の辺に代謝産物番号が  $m$  で係数が  $n$  の代謝産物分子があれば、その反応の基質の代謝産物番号の配列に  $m$  が  $n$  回現れる。生成物についても同様である。仮想原料反応 *source reaction* の基質の代謝産物番号の配列、仮想目的反応 *target reaction* の生成物の代謝産物番号の配列は空配列である。

## 2.3 EFM 型経路の化学量論的な代謝ネットワーク構造算出の概要

EFM 型経路の化学量論的な代謝ネットワーク構造に存在する反応ノードは、EFM 型経路の反応番号配列の要素と 1 対 1 に対応する。各反応ノードは、その反応の基質に相当するすべての代謝産物ノードそれぞれからのエッジが入る地点であるとともに、その反応の生成物に相当するすべての代謝産物ノードそれぞれに向けてのエッジが出る地点でもある。すなわち、各反応ノードには、代謝産物（基質）が入るエッジと（エッジを介して入る）代謝産物（基質）の組み合わせの集合と、代謝産物（生成物）が出るエッジと（エッジを介して出る）代謝産物（生成物）の組み合わせの集合が対応する。EFM 型経路の化学量論的なネットワーク構造に含まれるすべての代謝産物ノードはそれぞれが、1 エッジを介して 1 反応ノードから出て、別の 1 エッジを介して別の 1 反応ノードに入る。したがって、EFM 型経路の化学量論的なネットワーク構造において、エッジを介して反応ノードから出る代謝産物ノードの集合を  $P$ 、エッジを介して反応ノードに入る代謝産物ノードの集合を  $S$  とすると、集合  $P$  と集合  $S$  は代謝産物ノードの集合としては同一の集合でなければならない。すなわち、集合  $P$  の要素（どの反応ノードから出るかにより区別される代謝産物ノード）は集合  $S$  の要素（どの反応ノードに入るかにより区別される代謝産物ノード）と同一ノードとして 1 対 1 に対応付けられる。 $P$  の特定の要素と  $S$  の特定の要素は、両方が同じ（種類の）代謝産物のノードである場合に限り対応付けできる。化学量論的な代謝ネットワーク構造はこの対応付けにより生成する。アルゴリズム A, B による EFM 型経

路の化学量論的な代謝ネットワーク構造算出においては、この対応付けから生成し得るすべてのネットワーク構造から *unique* なものを見出すことを行うが、アルゴリズム A とアルゴリズム B では、これを異なる方法で行う。

### 2.3.1 アルゴリズム A

アルゴリズム A は、基本アルゴリズム部分と異同判定アルゴリズム部分から構成される。アルゴリズム A の基本アルゴリズム [14] 部分においては、ネットワークに含まれる代謝産物（の種類）ごとに、その（種類の）代謝産物の代謝産物ノードを要素とする  $P$  の部分集合と  $S$  の部分集合を作成し、両部分集合の要素を任意の組み合わせで対応付ける（同一とみなす）ことを行い、EFM 型経路の、すべての可能な、化学量論的な代謝ネットワーク構造を算出する。ネットワーク内に代謝産物が  $m_1$  から  $m_k$  の  $k$  種類あり、代謝産物番号が  $m_j$  ( $j$  は  $k$  以下の自然数) の代謝産物ノードの個数が  $n(j)$  であるとする、代謝産物番号が  $m_j$  の代謝産物ノードを要素とする  $P$  の部分集合と  $S$  の部分集合の要素間の対応付けは  $n(j)!$  通りある。したがって、 $P$  と  $S$  の要素間の可能な対応付け（化学量論的な代謝ネットワーク構造）が、全部で  $n(1)! \times n(2)! \times n(3)! \times \dots \times n(k)!$  通り得られる。 $P$  と  $S$  の要素間の可能な対応付けを 1 つ求めることは、 $P$  と  $S$  の要素間の 1 つの可能な対応付け  $x$  に対して、 $x$  で対応付けられた要素（ $P$  の要素に対応付けられた  $S$  の要素、 $S$  の要素に対応付けられた  $P$  の要素の何れか）の置換を 1 つ求めることと同等である。基本アルゴリズム部分に引き続く異同判定アルゴリズム [14] 部分では、基本アルゴリズム部分で得られたネットワーク構造の異同判定を行う。

### 2.3.2 アルゴリズム B

アルゴリズム B では、最初に、“ $P$  の特定の要素と  $S$  の特定の要素は、両方が同じ（種類の）代謝産物のノードである場合に限り対応付けできる”という条件（“対応付け条件”）を満たす、 $P$  の要素と  $S$  の要素の対応付けを 1 つ求める。この対応付けを  $x$  と呼び、 $x$  に対応するネットワーク構造を空集合  $U$  に要素として入れる。 $P$  の要素と  $S$  の要素の可能な対応付けは、すべてが、 $x$  により対応付けられた要素（ $P$  の要素に対応付けられた  $S$  の要素、 $S$  の要素に対応付けられた  $P$  の要素の何れか）の置換により得られる。以下、対応付け  $x$  により対応付けられた要素を置換することを、“ $x$  を置換する”（“ $x$  の置換”）、対応付け  $x$  により対応付けられた要素を互換することを“ $x$  を互換する”（“ $x$  の互換”）と呼ぶ。“ $x$  の置換”と“ $x$  の互換”は、それぞれ、“対応付け  $x$  を置換することにより得られた対応付け”、“対応付け  $x$  を互換することにより得られた対応付け”の意味でも用いる。互換が置換に含まれることから、“ $x$  の互換”は“ $x$  の置換”でもある。数学的に、任意の置換は、互換の組み合わせで表現できるので、 $x$  に互換（“対応付け条件”）を満たす、対応付けられた要素の互換）を繰り返し（ $x$  に対して可能な 1 回の互換をすべて行い、得られた  $x$

の置換それぞれに対して可能な1回の互換をすべて行い、さらに、得られたxの置換それぞれに対して可能な1回の互換をすべて行うことを繰り返す、xの置換を生成する。その際、各互換でxの置換yが得られるごとに、そのxの置換(Pの要素とSの要素の対応付け)yに対応するネットワーク構造nを求めて、ネットワーク構造の異同判定を行い、次のルールに従って、xの置換yの互換をやめるかyに次の互換を行うかを決める。

- nがUの要素であれば、xの置換yの互換をやめる。
- nがUの要素でなければ、nをUに要素として入れ、xの置換yに次の互換を行う。

互換によりUに含まれない新たなネットワーク構造nが生成されなくなるまで互換を続ける。EFM型経路のすべての可能な化学量論的な代謝ネットワーク構造がUの要素として得られる。

Uの要素に前述の手順で得られる順に番号を付し、i番目に得られたUの要素をU(i)と表す。U(i)は、xの特定の置換に対応するネットワーク構造として求まるものである。このU(i)に対応するxの特定の置換をx(i)で表す。また、xの置換yに対応するネットワーク構造をn(y)と表す。これらの記法によれば、 $U(i) = n(x(i))$ ,  $U(1) = n(x(1))$ ,  $x = x(1)$ である。

一般に、xの置換a, bに対して、

$$n(a) = n(b) \text{ならば } \{n(z)|z \text{ は } a \text{ の互換}\} = \{n(z)|z \text{ は } b \text{ の互換}\}$$

が成り立つ(“n(a) = n(b)のときn(a)のノードとn(b)のノードがnode content(後述)とネットワーク内のトポロジカルな位置が一致するよう1対1に対応付けできる”ことから“n(a) = n(b)であるときには{n(z)|zはaの互換}の要素と{n(z)|zはbの互換}の要素が1対1に付けできる”ことが証明できる)。したがって、前述の手順で最終的に得られるUとxの置換yに対してn(y) ∈ Uならば、n(y) = U(m)となるmが存在してn(y) = n(x(m)) = U(m) ∈ Uであり、

$$\{n(z)|z \text{ は } y \text{ の互換}\} = \{n(z)|z \text{ は } x(m) \text{ の互換}\} \subset U$$

が成り立つ。すなわち、xの置換yに対応するネットワーク構造がUに含まれていれば、yの互換に対応するネットワーク構造すべてがUに含まれる。x = x(1) (xの恒等置換)に対応するネットワーク構造U(1)がUに含まれているので、数学的帰納法により、xの任意の置換(互換の組み合わせで表される)に対応するネットワーク構造はすべてがUに含まれる。

## 2.4 EFM型経路の化学量論的な代謝ネットワーク構造算出のためのアルゴリズムBの補助関数

アルゴリズムBで使う補助関数には、calc\_sr\_rp, exist\_route, efm\_net\_transpositionがある。以下の記述では、配列Vのj番目の要素を“V[j]”, 配列Vの要素数を“length(V)”と表す記法を用いる。

### 2.4.1 calc\_sr\_rp

calc\_sr\_rpでは、EFM型経路の反応番号配列R, EFM型経路に含まれる各反応の情報(反応番号, 基質の代謝産物番号の配列, 生成物の代謝産物番号の配列)から, sr, rp, node\_list, cvniを算出する。srとrpは、EFM型経路の可能な化学量論的な代謝ネットワーク構造のうちの1つにおけるノード間のつながり(エッジ)情報を保持する2列の行列であり、srの行は代謝産物ノードから反応ノードに至るエッジと1対1対応し、rpの行は反応ノードから代謝産物ノードに至るエッジと1対1対応する。node\_listは、各ノードのnode number(ノード番号), node content(代謝産物番号または反応番号), node type(1: 代謝産物ノード, 2: 反応ノード)を格納する行列である。cvniは、srとrpの行番号の配列を要素とする配列である。srとrpは、各エッジの一方の端である代謝産物ノードの列(srでは1列目, rpでは2列目)の代謝産物種(代謝産物番号)ごとに行ソートされており、srの1列目の代謝産物ノードの代謝産物種(代謝産物番号)は、rpの2列目の代謝産物ノードの代謝産物種(代謝産物番号)と一致する。ネットワークに含まれる1つの特定の代謝産物種の代謝産物ノードを一方の端とするエッジに対応するsrとrpの行番号(srとrpに共通)すべてからなる配列(リスト)が、配列cvniの1要素として格納される。srをrpの上に積み重ねると、EFM型経路の化学量論的な代謝ネットワーク構造におけるノード間のつながりを示すエッジリストである行列CM\_sが得られる。逆にCM\_sからsrとrpを得ることもできる。EFM型経路の化学量論的な代謝ネットワーク構造それぞれは、node\_listとCM\_sの対として表現される。以下に、具体的な手順を示す。nR, nM, snr, rpnを空行列、iを1として計算を開始する。

**Step 1.** node\_number = i, 反応R[i]の基質の代謝産物番号の配列をs, jを1とする。snrに[s[j], node\_number]を新たな行として追加してjを1増加させる操作を、操作前のjがlength(s)以下である間、繰り返す。

**Step 2.** node\_number = i, 反応R[i]の生成物の代謝産物番号の配列をp, jを1とする。rpnに[node\_number, p[j]]を新たな行として追加してjを1増加させる操作を、操作前のjがlength(p)以下である間、繰り返す。

**Step 3.** node\_number = i, node\_content = R[i], node\_type = 2としてnRに[node\_number, node\_content, node\_type]を新たな行として追加する。

**Step 4.** 引き続き、Step 1, Step 2, Step 3の操作を、2からlength(R)まで1ずつiを増加させながら繰り返す。

**Step 5.** snrの行を、1列目の代謝産物番号が小から大に並ぶようにソートする。rpnの行を、2列目の代謝産物番号が小から大に並ぶようにソートする。ソート後のsnrの1列目とソート後のrpnの2列目が一致することを確認し、snrをソート後のsnrで、rpnをソート後のrpnで置き換える。

その後 M = (snr の 1 列目),  $i = 1$  とする.

**Step 6.**  $\text{node\_number} = \text{length}(R) + i$ ,  $\text{node\_content} = M[i]$ ,  $\text{node\_type} = 1$  として nM に [node\_number, node\_content, node\_type] を新たな行として追加する.

**Step 7.** 引き続き, Step 6 の操作を, 2 から length(M) まで 1 ずつ  $i$  を増加させながら繰り返す. 最終的に得られる nM の上に nR を積み重ねて得られる行列 (行数が  $\text{length}(R) + \text{length}(M)$ , 列数が 3) を node\_list とする.

**Step 8.** snr の 1 列目 (代謝産物番号の列) を nM の 1 列目 (node number の列) で置き換えたものを sr とする. rpn の 2 列目 (代謝産物番号の列) を nM の 1 列目 (node number の列) で置き換えたものを rp とする. Step 8 で用いる nM は, Step 7 で最終的に得られる nM である.

**Step 9.** M に格納されている代謝産物番号を unique 化した配列を uM とする. cvni を配列とし, 代謝産物番号  $uM[k]$  と等しい M の要素の要素番号すべてを  $cvni[k]$  に格納する.  $cvni[k]$  に格納される M の要素番号は,  $uM[k]$  が snr の 1 列目 (= rpn の 2 列目) に現れる snr (rpn) の行の行番号, node content が  $uM[k]$  である代謝産物ノードが sr の 1 列目 (= rp の 2 列目) に現れる sr (rp) の行の行番号と一致する.

#### 2.4.2 exist\_route

exist\_route は, 特定のネットワーク構造 (node\_list と CM\_s の対として与えられる) が, 与えられたネットワーク構造の集合 (後述の cnode\_list, cCM\_s として与えられる) に含まれるかどうかを, 異同判定アルゴリズム [14] により判定する. cnode\_list は node\_list を要素とする配列, cCM\_s は CM\_s を要素とする配列であり, cnode\_list と cCM\_s の組がネットワーク構造の集合を表す. 1 つのネットワーク構造に対応する対となった node\_list と CM\_s は, それぞれ, 配列 cnode\_list と配列 cCM\_s の同じ要素番号の位置に格納される.

#### 2.4.3 efm\_net\_transposition

efm\_net\_transposition においては, node\_list, sr, rp, cvni から, cCM\_s, csr, crp を算出する. cCM\_s, csr, crp および cnode\_list を空配列として計算を開始する. sr の  $cvni[k]$  行 1 列の要素に対する可能な 1 回の互換をすべて行うことを, 1 以上  $\text{length}(cvni)$  以下の整数  $k$  に対して行う. sr の各互換には, CM\_s (互換後の sr である srw と rp から作られる) と node\_list の対として記述されるネットワーク構造が対応する. “sr の  $cvni[k]$  行 1 列の要素に対する可能な 1 回の互換をすべて行うことを, 1 以上  $\text{length}(cvni)$  以下の整数  $k$  に対して行う” 際, sr の各互換に対応するネットワーク構造がその時点の cnode\_list と cCM\_s で示されるネットワーク構造の集合に含まれるかを exist\_route により判定し, 含まれていない場合は, srw (互換後の sr), rp, node\_list, CM\_s (srw と rp から作られる) を, それぞれ, 配列 csr と配列 crp, 配列 cnode\_list と配列 cCM\_s に新たな同じ番号の要素として格納する.

## 2.5 EFM 型経路の化学量論的な代謝ネットワーク構造算出のためのアルゴリズム B

EFM 型経路の反応番号配列 R, EFM 型経路に含まれる各反応の情報 (反応番号, 基質の代謝産物番号の配列, 生成物の代謝産物番号の配列) から, EFM 型経路の化学量論的な代謝ネットワーク構造の node\_list (すべてのネットワーク構造に共通) と cCM\_s (各ネットワーク構造のエッジリスト CM\_s を要素として格納する配列) を算出する.

**Step 1.** calc\_sr\_rp により, EFM 型経路の反応番号配列 R, EFM 型経路に含まれる各反応の代謝産物レベル情報から, sr, rp, node\_list, cvni を算出する.

**Step 2.** csr, crp, cCM\_sx, cnode\_list を (この Step では) それぞれ 1 要素からなる配列とし,  $csr[1]$  に sr を,  $crp[1]$  に rp を,  $cCM\_sx[1]$  に sr と rp からなる CM\_s を,  $cnode\_list[1]$  に node\_list を入れる.

**Step 3.** 配列 csrz, 配列 crpz に対して,  $csrz = csr$ ,  $crpz = crp$  とした後に, csr と crp を空配列,  $i$  を 1 とする.

**Step 4.**  $sr = csrz[i]$ ,  $rp = crpz[i]$  とし, efm\_net\_transposition により, node\_list, sr, rp, cvni から, cCM\_s, csr, crp を算出し,  $j$  を 1 とする.

**Step 5.**  $cCM\_sz = cCM\_sx$  とする.  $j$  が  $\text{length}(cCM\_s)$  以下である間, 下記の操作を繰り返す.

$CM\_s = cCM\_s[j]$  とし, exist\_route により node\_list と CM\_s の対で表されるネットワーク構造が cnode\_list と cCM\_sz の組で与えられるネットワーク構造の集合に含まれるかどうかを判定し, 含まれていればそのネットワーク構造に対応する cCM\_sz の成分を除いて  $j$  を 1 増加させ, 含まれていなければ, csr に  $csr[j]$  を, crp に  $crp[j]$  を, cCM\_sx に CM\_s を, cnode\_list に node\_list を新たな要素として格納して  $j$  を 1 増加させる.

$j$  が  $\text{length}(cCM\_s)$  より大きい (大きくなった) 場合は, 次の Step 6 に進む.

**Step 6.**  $i$  を 1 増加させる.

**Step 7.**  $i$  が  $\text{length}(csrz)$  以下である間, Step 4 から Step 6 の一連の操作を繰り返す.  $i$  が  $\text{length}(csrz)$  より大きい (大きくなった) 場合は, 次の Step 8 に進む.

**Step 8.** csr が空配列でない間, 一連の Step 3 から Step 7 の操作を繰り返す. csr が空配列である (空配列になった) 場合は,  $cCM\_s = cCM\_sx$  とし, 計算を終了する.

## 2.6 EFM 型経路の化学量論的な代謝ネットワーク構造算出の高速化のための付加ステップ

アルゴリズム A では, 基本アルゴリズム [14] 部分において, アルゴリズム B では, calc\_sr\_rp の step において cvni が算出され, アルゴリズム A とアルゴリズム B の何れにおいても, cvni に要素として格納されている行番号の配列に基づき置換 (アルゴリズム B では互換のみ) を行う. しかし, 置換を行っても異なるネットワーク構造を生成しないことが明らかな cvni の要素がある. 付加ステップにおいて

は, `truncate_cvni` により, このような要素を `cvni` から除く.

具体的に述べると, `truncate_cvni` は, `node_list`, `sr`, `rp` の情報を用いて, `cvni` から, (単独の) 要素 (=配列) に含まれる行番号すべてに対応する `sr` の部分行列, または, `rp` の部分行列に現れる反応ノード (反応ノード番号) が 1 種類であるような要素を除き `cvni` を `truncate` する.

付加ステップを含むアルゴリズム A とアルゴリズム B においては, `truncate_cvni` により `truncate` された `cvni` に要素として格納されている行番号の配列に基づき置換 (アルゴリズム B では互換のみ) を行う.

### 3. 計算実験の結果と考察

糖質代謝モデルネットワーク [6] のペントースリン酸経路に含まれる 2 種類の `transketolase` 反応・`transaldolase` 反応・`ribulose-5-phosphate 3-epimerase` 反応の計 4 反応 (文献 [6] のプロセス番号 26, 28, 29, 30) の逆反応と `ribose-5-phosphate ketoisomerase` 反応 (文献 [6] のプロセス番号 27) の組み合わせ (計 5 反応, 延べ 7 反応) により化学量論的に 2 分子の `fructose 6-phosphate` と 1 分子の `glyceraldehyde 3-phosphate` から 3 分子の `ribose 5-phosphate` を生成する反応収支式ができる. これは 5 反応 (延べ 7 反応) からなる EFM 型経路である. これを EFM 型経路 1 と呼ぶ. EFM 型経路 1 に含まれる反応を, それぞれの数を 2 倍にして組み合わせると, 化学量論的に 4 分子の `fructose 6-phosphate` と 2 分子の `glyceraldehyde 3-phosphate` から 6 分子の `ribose 5-phosphate` を生成する反応収支式ができる. これは 10 反応 (延べ 14 反応) からなる EFM 型経路である. これを EFM 型経路 2 と呼ぶ.

1 で述べたように, EFM 型経路 1 に炭素原子部分の完全原子レベルマッピングを行うと, `cycle` を含まない経路と `cycle` 含む経路に対応する 2 種類のマッピングが得られた [9,10]. 前稿 [14] では原子レベルの情報を用いることなく, 基本アルゴリズムとそれに引き続く異同判定アルゴリズム (本稿のアルゴリズム A) により “化学量論的に釣り合った代謝産物レベルネットワーク構造” (“化学量論的な代謝ネットワーク構造”) である “`cycle` を含まない経路” と “`cycle` 含む経路” を算出した. 本項の計算実験の結果としては, EFM 型経路 1, EFM 型経路 2 に 4 種類のアルゴリズム (アルゴリズム A, アルゴリズム A+付加ステップ, アルゴリズム B, アルゴリズム B+付加ステップ) を適用した結果あるいは適用した場合に必要な計算量を述べる. 計算は Matlab と GNU Octave 上で行った.

EFM 型経路 1 に 4 種類のアルゴリズムを適用すると, どのアルゴリズムでも [9,10,14] と同じく, アルゴリズムに含まれる異同判定の結果 2 種類の `unique` な経路が算出された. アルゴリズム A, アルゴリズム A+付加ステップ, アルゴリズム B, アルゴリズム B+付加ステップにおいて, それぞれ, 生成された異同判定前の構造 (置換) の数は,

96, 8, 14, 6, 行われた異同判定アルゴリズム [14] の回数は, 142, 10, 16, 8, Matlab による計算時間は, 約 0.6 秒, 約 0.05 秒, 約 0.06 秒, 約 0.04 秒, GNU Octave による計算時間は, 約 5 秒, 約 0.4 秒, 約 0.5 秒, 約 0.3 秒であった (Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz 1.99GHz).

EFM 型経路 2 に 2 種類のアルゴリズム (アルゴリズム B, アルゴリズム B+付加ステップ) を適用すると, どちらのアルゴリズムでもアルゴリズムに含まれる異同判定の結果 8 種類の `unique` な経路が算出された. アルゴリズム B, アルゴリズム B+付加ステップにおいて, それぞれ, 生成された異同判定前の構造 (置換) の数は, 328, 160, 行われた異同判定アルゴリズム [14] の回数 (括弧内 `exist_route` の実行回数) は, 542 (360), 612 (192), Matlab による計算時間は, 約 44 秒, 約 26 秒, GNU Octave による計算時間は, 約 454 秒, 約 270 秒であった (Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz 1.99GHz).

理論的には 2.3.1 に示した計算により, EFM 型経路 2 に 2 種類のアルゴリズム, アルゴリズム A, アルゴリズム A+付加ステップを適用すると, 基本アルゴリズム部分で異同判定前の構造 (置換) を, それぞれ, 955514880, 55296 の個数だけ生成する. アルゴリズム A+付加ステップの基本アルゴリズム部分の所要時間は, Matlab と GNU Octave でそれぞれ約 41 秒, 約 205 秒であり, アルゴリズム A の基本アルゴリズム部分の所要時間は, その 17280 (=955514880÷55296) 倍であると推定される. アルゴリズム A, アルゴリズム A+付加ステップの異同判定アルゴリズム部分では, それぞれ, 955514880, 55296 の構造から `unique` な構造を求めると, 異同判定アルゴリズムの必要実行回数は相当な数となる. したがって, EFM 型経路 2 にアルゴリズム A, アルゴリズム A+付加ステップ (特にアルゴリズム A) を適用すると膨大な計算時間が必要となる.

以上から, 置換により可能なネットワーク構造をすべて求めてからネットワーク構造の異同判定を行う (アルゴリズム A) 代わりに, 置換を互換の繰り返しにより生成して置換が生成することに置換に対応するネットワーク構造の異同判定を行う (アルゴリズム B) こと, および, 明らかに同じネットワーク構造を生成する置換を省略する (付加ステップ) ことが, EMU 型経路の化学量論的に釣り合った代謝産物レベルネットワーク構造 (化学量論的な代謝ネットワーク構造) の算出の高速化に有効である.

### 参考文献

- [1] Arita, M. *In silico* atomic tracing by substrate-product relationship in *Escherichia coli* intermediary metabolism. *Genome Res.* **2003**, *13*, 2455–2466. DOI: 10.1101/gr.1212003
- [2] Latendresse, M.; Kruppenacker, M.; Karp, P.D. Optimal metabolic route search based on atom mappings. *Bioinformatics* **2014**, *30*, 2043–2050. DOI: 10.1093/bioinformatics/btu150
- [3] Heath, A.P.; Bennett, G.N.; Kavraki, L.E. Finding metabolic pathways using atom tracking. *Bioinformatics* **2010**, *26*, 1548–1555.

DOI: 10.1093/bioinformatics/btq223

- [4] Pitkänen, E.; Jouhten, P.; Rousu, J. Inferring branching pathways in genome-scale metabolic networks. *BMC Syst. Biol.* **2009**, *3*, 1–22. DOI: 10.1186/1752-0509-3-103
- [5] Ohta, J. Connectivity matrix method for analyses of biological networks and its application to atom-level analysis of a model network of carbohydrate metabolism. *IEE Proc. Syst. Biol.* **2006**, *153*, 372-374. DOI: 10.1049/ip-syb:20060018
- [6] Ohta, J. Single-atom tracing in a model network of carbohydrate metabolism and pathway selection. *IPJS Transactions on Bioinformatics* **2018**, *11*, 1-13.
- [7] Schuster, S.; Hilgetag, C. On elementary flux modes in biochemical reaction systems at steady state. *J. Biol. Syst.* **1994**, *2*, 165-182.
- [8] Schilling, C. H.; Palsson, B.O. The underlying pathway structure of biochemical reaction networks. *Proc. Nat. Acad. Sci. USA* **1998**, *95*, 4193-4198.
- [9] 太田 潤 代謝ネットワークにおける elementary flux mode 型経路の完全原子レベルマッピング. 情報処理学会研究報告, **2019**, Vol. 2019-BIO-59, No. 2, 1-6.
- [10] 太田 潤 Elementary flux mode 型代謝経路の完全原子レベルマッピングの高速化. 情報処理学会研究報告, **2019**, Vol.2019-BIO-60, No. 6, 1-7.
- [11] 太田 潤 対称性のある代謝産物を通る elementary flux mode 型代謝経路の完全原子レベルマッピング. 情報処理学会研究報告, **2020**, Vol.2020-BIO-61, No. 9, 1-8.
- [12] Acuña, V.; Milreu, P.V.; Cottret, L.; Marchetti-Spaccamela, A.; Stougie, L.; Sagot, M.F. Algorithms and complexity of enumerating minimal precursor sets in genome-wide metabolic networks. *Bioinformatics* **2012**, *28*, 2474-83.
- [13] Ravikrishnan, A.; Nasre, M.; Raman, K. Enumerating all possible biosynthetic pathways in metabolic networks. *Sci. Rep.* **2018**, *8*, 9932. DOI: 10.1038/s41598-018-28007-7
- [14] 太田 潤 Elementary flux mode 型代謝経路の化学量論的に釣り合った代謝産物レベルネットワーク構造の算出. 情報処理学会研究報告, **2020**, Vol.2020-BIO-62, No. 3, 1-6.