

Elementary flux mode 型代謝経路の 化学量論的な代謝ネットワーク構造の算出： 算出されたネットワーク構造間の異同判定の高速化

太田潤^{†1,a)}

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”を考えることができる。この反応の集合としての“経路”を elementary flux mode 型経路 (EFM 型経路) と呼んでいる。特定の EFM 型経路を定義する、経路に含まれる反応に関する情報 (どの反応がいくつ含まれるか) から、その 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 (化学量論的な代謝ネットワーク構造) を求めることは、1 つの数学的な問題ととらえられる。そして、この問題を解くことで求まる化学量論的な代謝ネットワーク構造は、経路に含まれる反応がどのような順序とつながりで働いて経路の原料分子群から目的分子群が生成するかを示す。前二稿において、EFM 型経路の化学量論的な代謝ネットワーク構造を、その EFM 型経路を定義する代謝産物レベルの反応情報から算出するためのアルゴリズムを報告した。前二稿の研究では、算出される経路のネットワーク構造の異同判定がアルゴリズム全体の計算速度に影響を与える可能性が考えられた。そこで、前二稿で用いた異同判定アルゴリズム (旧異同判定アルゴリズム) の高速化を試みた。異同判定する 2 つのネットワーク構造におけるつながりの一致確認を、旧異同判定アルゴリズムでは、ネットワーク構造全体のみで行っていたが、本稿の高速化の試みでは、これに、部分ネットワーク構造レベルでのつながりの簡易一致確認 (部分ネットワーク構造に特有の指標に基づく) の段階を加えた。部分ネットワーク構造レベルでのつながりの簡易一致確認の段階の追加は、高速化に非常に有効であった。

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

Calculation of stoichiometric metabolic network structure of elementary flux mode type metabolic pathway: Acceleration of difference judgement between calculated network structures

JUN OHTA^{†1,a)}

1. はじめに

代謝ネットワークの主要な機能の 1 つである生理活性物質生成のための“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 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] が研究されてきた。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。

EFM 型経路においては、経路を定義する反応の集合に含まれる反応の情報から経路の収支式が求まり、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、反応の集合自体は、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) を示さない。

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 [9,10]、

EFM 型経路の化学量論的に釣り合った (化学量論的な) 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

^{†1} 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

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。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] などがあるが、前者の手法では生合成経路の算出はできず、後者の手法で算出される生合成経路は、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、化学量論的な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異なる原子レベルマッピングが対応すると考えられるので、EFM 型経路の化学量論的な代謝産物レベルネットワーク構造を、EFM 型経路を構成する反応の原子レベル情報を用いる 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。そして、特定の EFM 型経路を定義する、経路に含まれる反応に関する情報 (どの反応がいくつ含まれるか) から、その EFM 型経路の化学量論的な代謝産物レベルネットワーク構造を求めることは、1 つの数理的な問題ととらえられる。前二稿 [14,15] では、この問題に取り組み、原子レベルの情報を用いることなく、EFM 型経路を定義する代謝産物レベルの反応情報のみから EFM 型経路の化学量論的な代謝産物レベルネットワーク構造を算出するためのアルゴリズムを報告した。

前二稿 [14,15] のアルゴリズムには、異同判定アルゴリズムの段階が含まれる。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必要があるからである。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、異同判定アルゴリズムが、化学量論的な代謝産物レベルネットワーク構造算出アルゴリズム全体の計算速度に影響を与える。そこで、前二稿で用いた異同判定アルゴリズム (旧異同判定アルゴリズム) の高速化を試みた。本稿では、異同判定する 2 つのネットワーク構造におけるネットワーク構造全体でのつながりの一致確認の前に、部分ネットワーク構造レベルでのつながりの簡易一致確認 (部分ネットワーク構造に特有の指標に基づく) の段階を加えることを含む、異同判定高速化のための 3 つの試みについて述べる。

本稿の計算実験では、原子レベルではなく、代謝産物レベルの代謝ネットワーク構造を扱ったので、以下において“化学量論的な代謝ネットワーク構造”、“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”、“化学量論的な代謝産物レベルネットワーク構造”は同義である。

2. 方法

化学量論的な代謝ネットワーク構造の算出の概要、本稿

で扱う異同判定アルゴリズム、その異同判定アルゴリズムの化学量論的な代謝ネットワーク構造算出アルゴリズムへの組み込みについて述べる。本稿で扱う“化学量論的な代謝ネットワーク構造”は、代謝産物と反応をノードとするネットワーク構造である。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。2.1 に述べる変換後の EFM 型経路においては、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。また、本稿では、配列 Z の j 番目の要素を“ $Z[j]$ ”と表す記法を用いる。

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 [14,15]

化学量論的な代謝ネットワーク構造の算出のために、経路を構成する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な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。

$source\ reaction: no\ metabolite(s) \rightarrow source(s)$

$target\ reaction: target(s) \rightarrow no\ metabolite(s)$

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 2.7 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 [14,15]

化学量論的な代謝ネットワーク構造の算出のために、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 [14,15]

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（“対応付け条件”）。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。“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 x とすると、 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。

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（置換）に対応して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。そのためのアルゴリズムとして、アルゴリズム A とアルゴリズム B がある [15]。アルゴリズム A では、“対応付け条件”を満たす置換に対応するネットワーク構造をすべて求め、ネットワーク構造の異同判定アルゴリズムにより *unique* なネットワーク構造を求める。アルゴリズム B では、“対応付け条件”を満たす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

異同判定アルゴリズムでネットワーク構造の異同判定を行う。

2.4 EFM 型経路の化学量論的な代謝ネットワーク構造の記述 [14,15]

EFM 型経路の化学量論的な代謝ネットワーク構造は、 CM_s と $node_list$ の組み合わせにより記述される。 CM_s は、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（エッジ）情報を保持する 2 列の行列であり、各行は、代謝産物ノードから反応ノードに至るエッジ、または、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。

2.5 EFM 型経路の化学量論的な代謝ネットワーク構造の異同判定に用いるネットワーク構造情報

ネットワーク構造内の経路のノード番号配列を格納した cMn 、経路の $node\ content$ 配列を格納した cMe 、経路の通し番号（経路番号）を格納した cn 、経路間のつながり情報を含む $cMap$ を CM_s 、 $node_list$ 、0 以上の整数 $number_of_Map$ から算出する。

2.5.1 cMn

異同判定するネットワーク構造内のトポロジカルに特徴的なノードをつなぐ経路を算出する。この経路の算出は、ネットワーク構造内のノードであって算出された経路の何れにも現れないものがないような“一定の方法”で行う。算出された経路は、経路長と経路のノード内容 ($node\ content$ と $node\ type$ の組み合わせ) により分類し、経路のノード番号配列を配列 cMn に格納する。 cMn の各要素は、同じ経路長・同じ“経路のノード内容配列”を持つ経路のノード番号配列の集合（配列）であり、個々の、経路のノード番号配列ではない。 cMn の要素は、各要素 (= 集合) を特徴づける経路長や“経路のノード内容配列”に関する一定の規則に基づく順序で配列される。異同判定される複数のネットワーク構造に対応する cMn の要素は、すべてが同じ規則に基づく順序で配列される。

ネットワーク構造内の経路算出のための“一定の方法”として、下記の Method 1 または Method 2 を用いた。

Method 1. Method 1 は、旧異同判定アルゴリズム [14] の Step 1, Step 2 を修正したものである。

ネットワーク構造において、“入り”がなく“出”があるノードのリスト s 、“入り”はあるが“出”がないノードのリスト d を作成する。変換後の EFM 型経路に対して得られる化学量論的な代謝ネットワーク構造は、入るエッジ (“入り”) のないノードが仮想原料反応 *source reaction* のノードのみ、出るエッジ (“出”) のないノードが仮想目的反応 *target reaction* のノードのみの有向グラフであるので、仮想原料反応 *source reaction* のノードのみがリスト s に、仮

想目的反応 target reaction のノードのみがリスト d に入る。s に含まれるノードを始点とし、かつ、d に含まれるノードを終点とする経路をすべて算出する。元のネットワーク構造の部分ネットワーク構造であって、ここまでの経路計算で使われたことがあるエッジすべてからなるネットワーク構造 u と、ここまでの経路計算で一度も使われていないエッジすべてからなるネットワーク構造 nu を得る。ネットワーク構造 u とネットワーク構造 nu の両方に含まれるノードのリスト sd を作成する。ネットワーク構造 nu において、sd に含まれるノードを始点とし、sd に含まれるノードを通過せず、sd に含まれるノード（終点）に到達する経路をすべて算出する。

Method 2. ネットワーク構造において、そのノードへの“入り”のエッジの数とそのノードから“出”のエッジの数のうち少なくとも一方が 1 でないようなノードのリスト s、を作成する。そして、s に含まれるノードを始点とし、s に含まれるノードを通過せず、s に含まれるノード（終点）に到達する経路をすべて算出する。元のネットワーク構造の部分ネットワーク構造であって、ここまでの経路計算で使われたことがあるエッジすべてからなるネットワーク構造 u と、ここまでの経路計算で一度も使われていないエッジすべてからなるネットワーク構造 nu を得る。ネットワーク構造 u とネットワーク構造 nu の両方に含まれるノードのリスト sd を作成する。ネットワーク構造 nu において、sd に含まれるノードを始点とし、sd に含まれるノードを通過せず、sd に含まれるノード（終点）に到達する経路をすべて算出する。

2.5.2 cMe

cMe は配列であり、cMn[i]の j 番目の要素として含まれる経路の node content 配列を cMe[i]に j 番目の要素として格納する。cMe[i]の要素数は cMn[i]の要素数と一致する。

2.5.3 cn

cMn の要素（集合、配列）の要素である経路（ノード番号配列）に通し番号を付す。この通し番号を経路番号と呼ぶ。cn は配列であり、cMn[i]に含まれる経路の経路番号の配列を cn[i]に格納する。

2.5.4 cMap

cMap は配列である。number_of_Map が 0 の場合は、cMap を空配列とする。number_of_Map が 1 以上の場合、経路番号の最大値を行数および列数とする正方形 Map を作成する。経路番号が i の経路を path(i)と表現し、Map の i 行 j 列の成分を、i と j が異なる場合は path(i)と path(j)の両方に含まれるノードの個数、i が j と等しい場合は 0 とする。number_of_Map 以下のすべての自然数 i に対して cMap[i]に Map の i 乗 を格納して cMap を算出する。

2.6 EFM 型経路の化学量論的な代謝ネットワーク構造の異同判定アルゴリズム

ネットワーク構造 a（構造 a）の CM_s, node_list, cMn,

cMe, cn, cMap を、それぞれ、CM_s_a, node_list_a, cMn_a, cMe_a, cn_a, cMap_a とする。また、ネットワーク構造 b（構造 b）の CM_s, node_list, cMn, cMe, cn, cMap を、それぞれ、CM_s_b, node_list_b, cMn_b, cMe_b, cn_b, cMap_b とする。cMn_a, cMe_a, cn_a, cMap_a と cMn_b, cMe_b, cn_b, cMap_b は同じ方式で算出する。すなわち、同じ Method（Method 1 と Method 2 の何れか）で経路を算出し、同じ規則に基づく順序で cMn_a と cMn_b（cMe_a と cMe_b, cn_a と cn_b）の要素を配列し、同じ number_of_Map で cMap_a と cMap_b を算出する。

構造 a のノード数・エッジ数と構造 b のノード数・エッジ数とが一致しない場合、構造 a と構造 b は異なると判定される。一致する場合は、cMe_a と cMe_b を比較する。

cMe_a と cMe_b が異なる場合、構造 a と構造 b は異なる判定される。cMe_a と cMe_b が同じである場合は、cMn_a の要素数（=cMn_b の要素数）以下のすべての自然数 i に対して cMn_a[i]に含まれる経路と cMn_b[i]に含まれる経路を対応付けて生成する（cMn_a の要素の要素である経路に cMn_b の要素の要素である経路を node content 配列が同じもの同士 1 対 1 に対応付けて生成する）、構造 a と構造 b のノード間の対応付けの中に、同一構造条件を満たすものが存在しないかを調べる。同一構造条件は“構造 a のノード a1, a2 に対応する構造 b のノードを b1, b2 とするとき、構造 a の任意の 2 ノード a1, a2 に対して、a1→a2 の場合には b1→b2 であるが、a1→a2 でない場合には b1→b2 でない”（“→”はエッジ）である。同一構造条件を満たすノード間の対応付けが存在すれば、構造 a と構造 b は同じと判定され、存在しなければ、構造 a と構造 b は異なる判定される。

構造 a と構造 b のノード間の対応付けの中に、同一構造条件を満たすものが存在するかどうかを調べるためには、ノード間の対応付けを生成する経路間の対応付け（cMn_a の要素の要素である経路と cMn_b の要素の要素である経路の対応付け）を、木構造上の root から leaf に至る道筋と考えて探索する。cMe_a と cMe_b が同じ場合は cn_a と cn_b が同じなので、以下では cn_a と cn_b を cn と表現する。木構造の深さ i の各ノードは、cMn_a[i]に含まれる経路に対する cMn_b[i]に含まれる経路の 1 対 1 の対応付け（経路番号 cn[i]の構造 a の経路に 1 対 1 対応する構造 b の経路を示す cn[i]の順列）に対応させる。すなわち、木構造の深さ i の各ノードに cn[i]の順列が対応する。本稿では“順列”を“ある配列の置換により得られる配列”という意味で用いる。木構造のノード x に対応する順列を、ノード x の順列と呼び、順列(x)と表現する。木構造の深さが j のノードは、それが leaf でなければ、それぞれが、cn[j+1]の順列の総数と同じ個数の子ノードを持つ。ある異なる 2 ノードが共通の親ノードの子ノードである場合、それら 2 ノードの順列は互いに異なる。木構造の leaf の深さは、すべて、cMn_a

の要素数 (= cMn_b の要素数) に等しい。

木構造のノード x に対して、木構造の $root$ から x に至る道筋を、ノード x への道筋と呼び、道筋(x)と表現する。道筋(x)には、道筋(x)上の $root$ 以外のすべてのノードの順列により指定される“構造 a の経路と構造 b の経路の 1 対 1 の対応付け”が対応する。“構造 a の経路と構造 b の経路の 1 対 1 の対応付け”は“構造 a の部分構造と構造 b の部分構造のノード間の対応付け”でもある。道筋(x)に対応する“構造 a の経路と構造 b の経路の 1 対 1 の対応付け”を経路対応(x)、“構造 a の部分構造と構造 b の部分構造のノード間の対応付け”を部分構造ノード対応(x)と呼ぶ。また、経路対応(x)で対応付けられた構造 a と構造 b の経路を、それぞれ、構造 a 経路(x)、構造 b 経路(x)と呼び、部分構造ノード対応(x)で対応付けられた構造 a と構造 b の部分構造を、それぞれ、構造 $a(x)$ 、構造 $b(x)$ と呼ぶ。 f を leaf とするとき、構造 $a(f)$ 、構造 $b(f)$ は、それぞれ、構造 a 、構造 b と一致し、部分構造ノード対応(f)は、構造 a 全体と構造 b 全体のノード間の対応付けを意味する。したがって、 cMn_a の要素の要素である経路に cMn_b の要素の要素である経路を $node\ content$ 配列が同じもの同士 1 対 1 に対応付けて生成する、構造 a と構造 b のノード間の対応付けの中に、同一構造条件を満たすものが存在するかどうかを調べることは、部分構造ノード対応(f)の中に、同一構造条件を満たすものが存在するかどうかを調べることである。そのための手順は、 $number_of_Map$ ($cMap_a$ および $cMap_b$ の要素数) が 0 である場合と 1 以上の整数である場合で異なる。

2.6.1 $number_of_Map$ が 0 である場合

$number_of_Map$ が 0 である場合は、上記の部分構造ノード対応(f)の中に、同一構造条件を満たすものが存在するかどうかを、直接、順番に調べる。ある部分構造ノード対応(f)が同一構造条件を満たせば、構造 a と構造 b は同じであると判定される。構造 a と構造 b の異同判定のみが目的であればここで計算を停止するが、同一構造条件を満たす“構造 a 全体と構造 b 全体のノード間の対応付け”をすべて求めることが目的であれば、部分構造ノード対応(f)をすべて調べる。また、部分構造ノード対応(f)をすべて調べなければ、構造 a と構造 b が異なると判定することはできない。

2.6.2 $number_of_Map$ が 1 以上の整数である場合

$number_of_Map$ が 1 以上の整数である場合は、木構造におけるノードを深さ優先探索と同じ順序で探索する。この探索においては、“簡易構造指標”が、構造 $a(x)$ と構造 $b(x)$ の間で一致していなければ、ノード x の子孫を探索しない。“簡易構造指標”は、構造 $a(x)$ に対しては、構造 $a(x)$ を構成する構造 a 経路(x)に含まれる経路間のつながりの有無のパターンであり、構造 $b(x)$ に対しては、構造 $b(x)$ を構成する構造 b 経路(x)に含まれる経路間のつながりの有無のパターンである。構造 $a(x)$ と構造 $b(x)$ は、それぞれ、構造 a と構造 b の部分ネットワーク構造であるので、この“簡易構

造指標”は、構造 a と構造 b の部分ネットワーク構造に特有の指標である。この“簡易構造指標”が構造 $a(x)$ と構造 $b(x)$ で不一致の場合は、ノード x の子孫の leaf ノード f に対する部分構造ノード対応(f)であって同一構造条件を満たすものが明らかに存在しない。以下、実際の計算手順を示す。構造 $a(x)$ の“簡易構造指標”としては、 $cMap_a$ からの情報をそのまま用い、構造 $b(x)$ の“簡易構造指標”としては、 $cMap_b$ からの情報を経路対応(x)により構造 $a(x)$ の“簡易構造指標”と比較可能な形に変換したものを用いる。

- Step 1.** $issame$ を 0 とする。 x を木構造の深さ 1 のノードとし、 $root$ のみを要素とする集合 V に x を新たな要素として入れる。 $ccMap_b$ を配列とし、 $ccMap_b[1]$ に $cMap_b$ を格納する。 Step 2 に進む。
- Step 2.** x の深さを $depth(x)$ と表現する、 t を 1 とする。 $cMap_b_temp$ を $ccMap_b[depth(x)]$ とする。 j を 1 とし、 Step 3 に進む。
- Step 3.** j が $number_of_Map$ より大きければ Step 4 に進む。 j が $number_of_Map$ 以下であれば、 M を $cMap_b_temp[j]$ とする。 M の $cn[depth(x)]$ に含まれる番号の行を、順列(x)により対応付けられた番号の行と入れ換える。次いで、 M の $cn[depth(x)]$ に含まれる番号の列を、順列(x)により対応付けられた番号の列と入れ換える。 $cMap_b_temp[j]$ を M とする。 $cn[depth(x)]$ に含まれる番号の最大値を m とする。 $cMap_a[j]$ の 1 から m 行目、1 から m 列目の部分行列が、 M の 1 から m 行目、1 から m 列目の部分行列と一致すれば j を 1 増加して Step 3 を再度行う。一致しなければ t を 0 とし、 Step 5 に進む。
- Step 4.** $ccMap_b[depth(x)+1]$ に $cMap_b_temp$ を格納する。 $depth(x)$ が cn の要素数と等しければ Step 6 に進む (x は leaf である)。 $depth(x)$ が cn の要素数と等しくなければ、 x の子ノードを新たな x とする。新たな x を集合 V に要素として入れ、 Step 2 に進む。
- Step 5.** そのノードの親ノードが集合 V の要素であるがそのノード自身は集合 V の要素でないようなノードの集合 C を求める。集合 C が空集合の場合は Step 7 に進む。集合 C が空集合でない場合は、集合 C に含まれるノードのうち x から最も近いものを新たな x とする。そして、新たな x を集合 V に要素として入れ、 Step 2 に進む。
- Step 6.** 部分構造ノード対応(x)が同一構造条件を満たせば、 $issame$ を 1 とする。構造 a と構造 b の異同判定のみが目的であれば、部分構造ノード対応(x)を記録して Step 7 に進む。同一構造条件を満たす部分構造ノード対応(x)をすべて求めることが目的であれば Step 5 に進む。本稿の計算実験においては、異同判定アルゴリズムを、異同判定のみを目的として使用した。
- Step 7.** $issame$ が 1 であれば、構造 a と構造 b は同じであると判定され、 $issame$ が 0 であれば、構造 a と構造 b は異なると判定される。

2.7 EFM 型経路の化学量論的な代謝ネットワーク構造算出アルゴリズムへの異同判定アルゴリズムの組み込み

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“構造情報非保持型アルゴリズム”と“構造情報保持型アルゴリズム”について述べる。

2.7.1 構造情報非保持型アルゴリズム

化学量論的な代謝ネットワーク構造の異同判定のためには、旧異同判定アルゴリズムと本稿の異同判定アルゴリズムのどちらにおいても、異同判定するネットワークの `CM_s`, `node_list` 以外の構造情報を `CM_s`, `node_list` から計算することが必要である。前二稿 [14,15] の EFM 型経路の化学量論的な代謝ネットワーク構造算出アルゴリズムは、何れも、算出されたネットワークの構造情報として `CM_s`, `node_list` のみを保持し、`CM_s`, `node_list` 以外の構造情報を異同判定の都度、異同判定アルゴリズムの中で計算するように設計していた。本稿の計算実験における“構造情報非保持型アルゴリズム”は、前稿 [15] の“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+付加ステップ”における異同判定アルゴリズムを本稿の異同判定アルゴリズム (2.5 に引き続く 2.6) に替えたものである。

2.7.2 構造情報保持型アルゴリズム

“構造情報非保持型アルゴリズム”では、異同判定のための `CM_s`, `node_list` 以外の構造情報を求めるために、全く同じ計算を何回も行うことがある。これを避けるため、前稿の 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 を改良した。改良型アルゴリズム B では、ネットワーク構造が算出されると `CM_s`, `node_list` 以外の構造情報 (2.5 に述べたもの) を計算して保持し、異同判定の際には、保持された構造情報を用いて本稿の異同判定アルゴリズムの 2.6 の部分のみを行う。本稿の計算実験における“構造情報保持型アルゴリズム”は、“改良型アルゴリズム B+付加ステップ”である。

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。n を自然数とするとき、EFM 型経路 1 に含まれる反応を、それぞれの数を n 倍にして組み合わせると、化学量論的に $2 \times n$ 分子の *fructose 6-phosphate* と n 分子の

glyceraldehyde 3-phosphate から $3 \times n$ 分子の *ribose 5-phosphate* を生成する反応収支式ができる。これは $5 \times n$ 反応 (延べ $7 \times n$ 反応) からなる EFM 型経路である。これを EFM 型経路 n と呼ぶ。前稿 [15] では、EFM 型経路 2 (n が 2 のときの EFM 型経路 n) の化学量論的な代謝ネットワーク構造として、8 種類の unique な経路が算出されることを述べた。本項の計算実験の結果としては、EFM 型経路 2 の化学量論的な代謝ネットワーク構造の計算時間が、構造情報非保持型アルゴリズムと構造情報保持型アルゴリズムのどちらを選択するか、Method 1 と Method 2 のどちらを選択するか、`number_of_Map` の値として 0, 1, 2, 3 のどれを選択するかという 3 つの要因によりどのように変化したかを調べた結果を述べる。原理的に、前稿 [15] で最速であった“アルゴリズム B+付加ステップ”と同等とみなされる、本稿で異同判定の高速化を企図して導入した、構造情報保持型アルゴリズム、Method 2、`number_of_Map` 値 1, 2, 3 (部分ネットワーク構造に特有の指標に基づく、部分ネットワーク構造レベルでのつながりの簡易一致確認) が有効であったかどうか、構造情報非保持型アルゴリズム - Method 1 - `number_of_Map` 値 0 の組み合わせの場合の計算時間との比較により確認できる。計算は Matlab 上で行った。

表 1 化学量論的な代謝ネットワーク構造の算出*

number_of_Map	計算時間 (秒)			
	構造情報非保持型		構造情報保持型	
	Method 1	Method 2	Method 1	Method 2
0	22.40	6.02	19.28	4.40
1	7.63	5.11	2.50	1.16
2	7.70	5.17	2.54	1.16
3	7.80	5.21	2.57	1.17

* EFM 型経路 2 の化学量論的な代謝ネットワーク構造を Matlab により計算した (Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz 1.99GHz)。

計算実験の結果を表 1 に示す。構造情報保持型アルゴリズム、Method 2、`number_of_Map` 値 1, 2, 3 の何れかを用いた場合は、どの組み合わせであっても計算時間が、構造情報非保持型アルゴリズム - Method 1 - `number_of_Map` 値 0 の組み合わせの場合の計算時間 22.40 秒よりも短かった。構造情報保持型アルゴリズム - Method 2 - `number_of_Map` 値 1 以上の組み合わせの場合の計算時間は、約 1.2 秒であり、この組み合わせから 3 要因のうちどの 1 要因を他のものに変更しても計算時間が 2 倍以上に増加した。この計算実験においては、構造情報保持型アルゴリズム、Method 2、1 以上の `number_of_Map` 値の採用が、異同判定の高速化に有効であったと言える。このうち少なくとも構造情報保

持型アルゴリズムと 1 以上の `number_of_Map` 値の採用は、原理的に、他の EFM 型経路においても有効であると考えている。1 以上の `number_of_Map` 値の採用は、異同判定アルゴリズムへの、部分ネットワーク構造レベルでのつながりの簡易一致確認（部分ネットワーク構造に特有の指標に基づく）の段階の追加を意味する。この計算実験の結果では、`number_of_Map` 値 1, 2, 3 の間に差が見られなかった。

構造情報保持型アルゴリズム - Method 2 - `number_of_Map` 値 1 を EFM 型経路 3 に適用したところ約 19 秒の計算時間で 26 種類の経路が算出された。EFM 型経路 `n` は `n` が増加するにつれてより多様な構造をとると推定される。本稿で述べた方法は、EFM 型経路 `n` を一般化した経路の構造の研究に有用であると考えられる。

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.; Kavasaki, 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. *IPSI 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.

- [15] 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算出の高速化. 情報処理学会研究報告, **2020**, Vol.2020-BIO-63, No. 1, 1-7.