

第2章

貿易商品分類 SITC から IO24 部門分類への変換

—変換エラーデータの処理—

黒子正人

はじめに

本書はアジア経済研究所の国際産業連関表作成プロジェクトが設定している国際産業連関表の24部門分類コード(以下、IO24と略記)にもとづく世界貿易マトリクスを作成することを目的としている。世界貿易マトリクス作成に先立ち、元データとなるアジア経済研究所の世界貿易統計データ:AID-XT(Ajiken Indicator of Developing economies: extended for Trade Statistics)の基礎データで使われているSITC各改訂版商品分類コード(以下SITCと略記)をIO24に変換する必要がある。SITCからIO24への変換は古河・野田による『標準国際貿易商品分類と産業分類の対応関係』(アジア経済研究所統計資料シリーズNo.80)にある改訂版対応表(SITC/IO24対応表)を利用する。

本章ではAID-XT基礎データのSITCに対して改訂版対応表を通してIO24へ変換する方法を紹介する。特に、この変換の際に対応表と一致しないAID-XT基礎データのSITCのデータ処理を中心に、変換処理の概要と具体的な処理手順を説明する。変換処理の方式として、均等配分方式および金額加重配分方式の2つを取り上げる。

本書の第2部にある表2の「世界貿易マトリクス:国際産業連関表24部門分類にもとづく時

系列取引額表」は本章の均等配分方式による方法を利用して作成されている。

1. 均等配分による変換

SITCをIO24に変換するには、まず、SITCをキーにしてSITC/IO24改訂版対応表(以下対応表と略記)をサーチし、該当するSITCがあった場合に、それに対応するIO24を割り当てる。ここでいう対応表とは、4桁レベルあるいは5桁レベルで構成されるSITCの基本分類(item)の一つに対して一つのIO24が対応するようなテーブルファイルである。該当するSITCがあった場合は出力データにIO24を転記する。このような変換処理ができたデータをパスデータとする。これは対応表のSITCとIO24の対応関係によって変換できる場合である。SITCに対応するIO24が一意に定まるため、変換処理としては何の問題も発生しない^(注1)。

例えば、SITC改訂第1版(SITC-R1)の0511を取り上げる。この対応表は、

(改訂版対応表)		(IO24への変換)	
SITC-R1	IO24	IO24	金額
04883	08	02	100
*0511	02		
0512	02		

と表され0511はIO24の02と対応している。

図1 エラーデータの例 (1) : SITC-R1 における 2422 の 100US\$ を IO24 部門分類へ変換

(対応表)		4 桁レベル分類コード			(IO24 への変換)	
SITC	IO24	SITC	IO24	f	IO24	金額
2421	04	2421	04	1	04	100
24221	04	*2422	04	2		
24222	04	2323	04	1		
24231	04					

(出所) 筆者作成

(注) 検索の対象となる SITC の 2422 を→の下で表示する。4 桁レベル分類コードである 2422 は対応表から得られた 4 桁レベル分類コードの対応関係の 2422 と一致し、この一致した SITC を*で示す。f は下位レベル分類コードの頻度を表す。

図2 エラーデータの例 (2) : SITC-R1 における 0530 の 100US\$ を IO24 部門分類へ変換

(対応表)		4 桁レベル分類コード			3 桁レベル分類コード			(IO24 への変換)	
SITC	IO24	SITC	IO24	f	SITC	IO24	f	IO24	金額
05209	02	0520	02	1	052	02	1	08	100
0532	08	0532	08	1	*053	08	8		
0533	08	0533	08	1	054	02	1		
0535	08	0535	08	1					
05361	08	0536	08	4					
05362	08	0539	08	1					
05363	08	0541	02	1					
05364	08								
0539	08								
0541	02								

(出所) 筆者作成

AID-XT 基礎データにある SITC の 0511 はこの対応表の中の SITC と直接一致するため、パステータである。対応表の中で一致した SITC を*で示す。0511 の取引額が 100US\$ とすれば、変換後の IO24 の 02 は 100US\$ である。

次に、SITC をキーにして SITC/IO24 対応表をサーチした結果、対応する分類コードが存在しないとき、すなわち、エラーになった場合であり、このようなデータをエラーデータという。これ以降が本処理の主な特徴となる。エラーとなるデータには様々なパターンがあるが、ここでそのうちのいくつかを取り上げて見てみたい。最初に取り上げるエラーデータの例 (1) は SITC-R1 における 2422 である。対応表の SITC

のままだと AID-XT 基礎データの SITC に対応しないが、対応表の SITC の末尾 1 桁を無視して 1 桁少ない分類コードでみた場合、AID-XT 基礎データの SITC とマッチするような場合である (図1 参照)。例 (1) の場合、対応表には SITC の 2422 にマッチするデータは存在しない。しかし対応表に SITC の上位桁レベルが 2422 に一致するデータが 2 件、24221 と 24222 が存在し、それらに対応する IO24 が一つの 04 に決定できる。この場合は、変換後出力データとして、SITC の 2422 に IO24 の 04 を対応させて取引額として 100US\$ を変換後出力データ (パステータ) に出力する。

例 (2) の場合は SITC-R1 における 0530 であ

図3 エラーデータの例 (3) : SITC-R1における2762の100US\$をIO24部門分類へ変換

(対応表)		4桁レベル分類コード			(IO24への変換)	
SITC	IO24	SITC	IO24	f	IO24	金額
2761	07	2761	07	1	07	75
27621	07	*2762	07	3	12	25
27622	07	*2762	12	1		
27623	07	2763	07			
27624	12					
2763	07					

(出所) 筆者作成

図4 エラーデータの例 (4) : SITC-R1における0540の100US\$をIO24部門分類へ変換

(対応表)		4桁レベル分類コード			3桁レベル分類コード			(IO24への変換)	
SITC	IO24	SITC	IO24	f	SITC	IO24	f	IO24	金額
0539	08	0539	08	1	053	08	1	02	82
0541	02	0541	02	1	*054	02	9	08	18
0542	02	0542	02	1	*054	08	2		
0544	02	0544	02	1	055	02	1		
0545	02	0545	02	1					
05461	08	0546	08	2					
05462	08	0548	02	5					
05481	02	0551	02	1					
05482	02								
05483	02								
05484	02								
05489	02								
0551	02								

(出所) 筆者作成

り、対応表にSITCの0530に一致するデータが存在しないのは前の例(1)と同じである。(図2参照)しかし今回は逆に、AID-XT基礎データのSITCの末尾1桁を無視して053としたときに対応表にSITCの上位桁レベル分類コードがこれに一致するデータが8件存在する。(対応表のSITCの末尾を1桁ないし2桁落として見る必要がある。)そして、それらに対応するIO24が一つの08に決定できる。この場合は、IO24の08にAIDXT基礎データを対応させる。

例(3)の場合はSITC-R1における2762であり、対応表にAID-XT基礎データのSITCの2762に対応するデータは存在しないが、対応表に

SITCの上位桁レベル分類コードが2762に一致するデータが4件存在する(図3参照)。しかし、これまでと異なり、それらは一つに決定できず、IO24の07のデータが3件、IO24の12のデータが1件となっている。このような場合はSITCとIO24の対応関係における頻度の比率で金額と数量を配分したデータを出力する。例えば例(3)の場合は、IO24の07の比率は、4分の3なので金額の配分は $100 \times 3/4 = 75$ となり、IO24の12の比率は4分の1なので金額の配分は $100 \times 1/4 = 25$ となる。この方式を均等配分方式と呼ぶのは、このように対応関係における頻度の比率で均等に配分することからきている。

図5 オリジナル対応表から拡張対応表 (2) までの作成過程

オリジナル対応表		拡張対応表 (1)		拡張対応表 (2)								
SITC-R1	IO24	SITC-R1	IO24	SITC-R1	01	02	03	...	08	...	20	合計
05461	08	0	08	0					1			1
		05	08	05					1			1
		054	08	054					1			1
		0546	08	0546					1			1
		05461	08	05461					1			1

(出所) 筆者作成

図6 拡張対応表 (2) から拡張対応表 (3) までの作成過程

拡張対応表 (2) の 05461				拡張対応表 (2) の 05489				拡張対応表 (3)								
SITC-R1	...	08	合計	SITC-R1	01	02	...	合計	SITC-R1	01	02	...	08	...	20	合計
0		1	1	0		1		1	0	9	2					11
05		1	1	05		1		1	05	9	2					11
054		1	1	054		1		1	054	9	2					11
0546		1	1	0548		1		1	0541	1	1					1
05461		1	1	05489		1		1	:							
									0548	5						5
									05461				1			1
									:							
									05489	1						1

(出所) 筆者作成

(注) 拡張対応表 (2) の 05461 から拡張対応表 (2) の 05489 までの表を積み重ねるようにして合計することで拡張対応表 (3) を作成する。

例 (4) の場合は SITC-R1 における 0540 であり、例 (3) と似ており、金額配分の原則は同じである (図4 参照)。AID-XT 基礎データの SITC の 054 までに一致する SITC 上位桁レベル分類コードをもつ対応表のデータが 11 件あるうち、9 件が IO24 の 02 に対応しており、2 件が IO24 の 08 に対応している。金額をそれぞれの頻度の比で配分する (小数点以下第 1 位を四捨五入する)。

1.1 具体的な処理手順

基本的には AID-XT 基礎データと対応表の SITC をキーとした表引き (テーブルサーチ) の

問題である。しかし、通常の表引きと異なり、それぞれのキーを変化させながら処理しなければならない。上記のエラーの例 (1) および (3) の場合は、AID-XT 基礎データ側の SITC は変化させずにキーにし、同時に対応表側の SITC は末尾 1 桁を落としたものをキーにして表引きする必要がある。一方、エラーの例 (2) と (4) の場合は、AID-XT 基礎データ側の SITC 末尾 1 桁を落としたものをキーにし、同時に対応表側の SITC は末尾 1 桁ないし 2 桁を落としたものをキーにしなければマッチしない。同時に双方の SITC の末尾を 1 桁ずつ落としてキーを作りながら表引きする方式は不可能ではないが、上記のようなさまざまなエラーのパターンにマッ

ちさせるためには多くの組合せを試行する必要があり、効率的に処理できない。そこで以下のような方法を採用した。

[1] 拡張対応表の作成処理：対応表側の SITC の末尾桁を 1 桁ずつ落とした全てのパターンを登録した表を最初に作成しておく。例えば、図 5 のようにオリジナルの対応表における SITC の 05461 が IO24 の 08 と対応している。これを 0、05、054、0546、05461 に分けて書き出し、対応する IO24 をそれぞれ 08 とする。これを以後、オリジナルの対応表の拡張版として使用するため、拡張対応表と略記する。

次に拡張対応表の IO24 を横方向への繰り返し 21 回の配列に展開する^(注2)。そして対応する IO24 の列と 21 番目の合計列に 1 を入れる(図 5)。この処理を各変換データに対して行った後、同一 SITC について、01 から 20 と合計の列を集計する。上記のエラー例 (4) の部分をこの方法で処理した結果を図 6 に例示する。ただしこれはあくまでもサンプルであり、0、05 などはその配下にあるコードが多数あるため、実際の拡張対応表の各値はこれよりも大きい値になる。

[2] オリジナル対応表による変換処理：ここからが実際の変換処理である。まず、第 1 段階として、拡張版ではないオリジナルの対応表と AID-XT 基礎データの表引きを行う。実際にプログラムが行う処理としては、オリジナルの対応表をメモリに全て読み込み、二分探索(バイナリサーチ)のサブプログラム(@bsch)を利用して AID-XT 基礎データの SITC でこの表の SITC をサーチすることにより表引き処理を行う。ここで前述のパスデータと変換できなかったエラーデータとが出力される。

[3] 拡張対応表による 1 回目の変換処理：次に [2] でエラーになったデータと [1] で作成した拡張対応表との表引きを行う。ここではまだ AID-XT 基礎データの SITC は末尾を削除するなどの加工は行わずそのままの形である。

一方拡張対応表の方はオリジナルの対応表の SITC を加工して、全ての桁レベルでの表引きが可能になっている。この処理により、前述のエラーの例 (1) と例 (3) のような場合、AID-XT 基礎データが変換されパスデータとして出力される。一方、エラーの例 (2) と例 (4) のような場合、AID-XT 基礎データの SITC が変わらなければマッチしないのでここではエラーデータとして出力される。

[4] 拡張対応表による 2 回目以降の変換処理：今までのステップでは AID-XT 基礎データの SITC を加工せずにマッチングを行ったが、このステップでは、AID-XT 基礎データの SITC を加工して拡張対応表との表引きを行う。ここでは読み込んだ AID-XT 基礎データの SITC を末尾から 1 桁ずつ削除しては表引きすることをマッチするまで do while ループで繰り返す。最悪の場合でも SITC の 1 桁レベルでは必ず拡張対応表に該当する SITC があってマッチするはずなので、この処理により全ての AID-XT 基礎データが IO24 に変換されてパスデータとして出力される。

[5] パスデータのソート・集計処理：これまでに出力されたパスデータをソートし集計する。

1.2 特徴

均等配分方式の短所を挙げると、実際の金額数量の実績との整合性がとれないことがありうることである。均等配分方式では対応関係の頻度の比率で配分する。そのため、前述のエラーデータの例 (3) の場合、IO24 の 07 に 75%を配分し、IO24 の 12 に 25%を配分する。ここで、SITC の 2762 のうち、27624 の実際の金額数量の実績によっては、コードの頻度による配分の妥当性が変わる。例えば、2762 の全体のうち、27624 が実際の金額で半数を占めるとしよう。すると、IO24 の 07 に 50%、12 に 50%の配分を

図7 AID-XT 基礎データと対応表から拡張対応表 (1) の作成過程

AID-XT 基礎データ					&	オリジナル対応表		→	拡張対応表 (1)					
RC	D	SITC	Y	金額		SITC IO24			RC	D	SITC	IO24	Y	金額
117310	1	0011	62	100		0011	03		117310	1	0011	03	62	100
117310	1	0011	63	200		0111	08		117310	1	0011	03	63	200
117310	1	0111	62	300					117310	1	0111	08	62	300
117310	1	0111	63	400					117310	1	0111	08	63	400

(出所) 筆者作成

(注) 報告国の 117310 はアジ研統一国コードの日本、輸出入区分の 1 は輸入、年は西暦年の下 2 桁を表す。相手国は省略している。& は AID-XT 基礎データと対応表を利用したと解釈する。

図8 拡張対応表 (2) から拡張対応表 (3) の作成過程

拡張対応表 (2)						→	拡張対応表 (3)									
RC	D	Y	SITC	IO24	金額		RC	D	Y	SITC	01	02	03	...	08	...
117310	1	62	0	03	100		117310	1	62	0			25		75	
117310	1	62	0	08	300		117310	1	62	00			100		0	
117310	1	62	00	03	100		117310	1	62	001			100		0	
117310	1	62	001	03	100		117310	1	62	0011			100		0	
117310	1	62	0011	03	100		117310	1	62	01			0		100	
117310	1	62	01	08	300		117310	1	62	011			0		100	
117310	1	62	011	08	300		117310	1	62	0111			0		100	
117310	1	62	0111	08	300		117310	1	63	0			25		75	
117310	1	63	0	03	200		117310	1	63	00			100		0	
117310	1	63	0	08	400		117310	1	63	001			100		0	
117310	1	63	00	03	200		117310	1	63	0011			100		0	
117310	1	63	001	03	200		117310	1	63	01			0		100	
117310	1	63	0011	03	200		117310	1	63	011			0		100	
117310	1	63	01	08	400		117310	1	63	0111			0		100	
117310	1	63	011	08	400											
117310	1	63	0111	08	400											

(出所) 筆者作成

(注) 表頭の 01...20 は IO24 の部門分類コードを表す。また、拡張対応表 (2) から同 (3) の作成は図 5 の方法と同じであるが、該当する箇所取引額が入り、SITC ごとに合計で除した比率が表示される。合計は 1 となるので省略する。

すべきだったのに前者には 25% 多く、また後者には 25% 少なく配分してしまったことになる。もちろんその逆もありうる。

長所としては、最終的に変換できないエラーデータが発生しないこと、すなわち、すべてが変換されることである。前述したように最終的にはエラーデータの SITC は 1 桁レベル分類コ

ードになるため、必ず IO24 に配分される。

2. 金額加重配分による変換

均等配分方式の欠点である、実績に見合わない配分をしてしまう点を是正できないか、ということから考案されたのが金額加重配分による

変換法である。金額加重配分方式と前述の均等配分方式とは、オリジナルの対応表では変換できなかった AID-XT 基礎データについてどのように処理していくかという点異なる。

均等配分による変換法では基本的に対応表を拡張してエラーを解消した。一方、金額加重配分による変換法では、最初にオリジナルの対応表を使って AID-XT 基礎データを変換するところは同じだが、その後、ここでパスしたデータから対応表を独自に作成し、そこで SITC ごとの対応する IO24 の頻度ではなく該当する金額の集計値によるウェイトをつけて配分する。この独自に作成する対応表は、これも均等配分とは異なり、キーとして報告国、輸出入区分、年、各桁レベルの SITC を持つ。SITC 以外の項目をキーとして持つ理由の一つは、AID-XT 基礎データではこれらの項目を無視して各桁レベルの SITC の金額ウェイトを求めることもできるがそれでは配分の情報として大まかすぎるということである。もう一つの理由は AID-XT 基礎データでは、これらの項目ごとに存在する SITC の組合せが異なるということである。

2.1 具体的な処理手順

[1] オリジナル対応表による変換処理：まず、オリジナルの対応表によって変換を行う。これは均等配分による変換法の [2] 第 1 回目の変換処理と全く同じ処理である。

[2] パスデータから拡張対応表の作成：次にパスデータとして出力されたデータから独自に対応表を作成する（以降ではこれを拡張対応表と略記する）。拡張対応表は、報告国 (RC)、輸出入区分 (D)、年 (Y)、各桁レベルの SITC、IO24 (20 個の配列) のようなフォーマットとなる。前述の理由から報告国、輸出入区分、報告年、各桁レベルの SITC がキーとなっている。このフォーマットを持つ拡張対応表を作成する

手順を以下にサンプル例をもとに説明する。

まず、パスデータの項目から上記のフォーマットの項目だけを選択する (図 7)。次に各桁レベルの SITC ごとにデータを出力してキーの順番に並び替える (図 8)。同一キーを 1 行にまとめる。IO24 を配列にし、各同一キー内での金額合計にしめる各 IO24 の金額の割合をパーセンテージで格納する。ここでは実際の金額を格納してその都度割合を算出するようにしてもよいが、AID-XT 基礎データの金額項目は 13 バイトを要するためデータ領域削減のためにパーセンテージとしている。また合計のための 21 番目の列は不要となる。これで拡張対応表が完成した。

[3] 拡張対応表での 1 回目の変換処理：ここからが実際の変換処理となる。[1] でエラーとなった AID-XT 基礎データと [2] で作成した拡張対応表とを読み込んでマッチングを行う。このときはまだ AID-XT 基礎データの SITC は加工せず元のままである。

なお、プログラムとしては均等配分方式と同じ機能をもつものでよいが、均等配分方式のプログラムは拡張対応表を全てメモリ上に読み込んでから、AID-XT 基礎データを 1 件ずつ読み込んで AID-XT 基礎データの各桁レベルの SITC が対応表にあるかどうかを探し、いわゆる

「表引き」の手法で書かれている。金額加重配分方式でもこのような「表引き」で処理することは論理的には可能だが、金額加重方式の拡張対応表のファイルはキーが多いため、必然的に容量が相当な大きくなる。計算機への負荷を考慮して、ここでは 2 つのファイルのマッチングの手法によるプログラムとした^(注3)。

[4] 拡張対応表での 2 回目以降の変換処理：ここからの変換処理は AID-XT 基礎データの SITC を一桁ずつ落としながら拡張対応表とのマッチングを行う。均等配分の [4] 拡張対応表による 2 回目以降の変換処理と機能的に同等であるが、[3] の処理と同様にファイルのマッ

図9 均等配分方式の流れ図

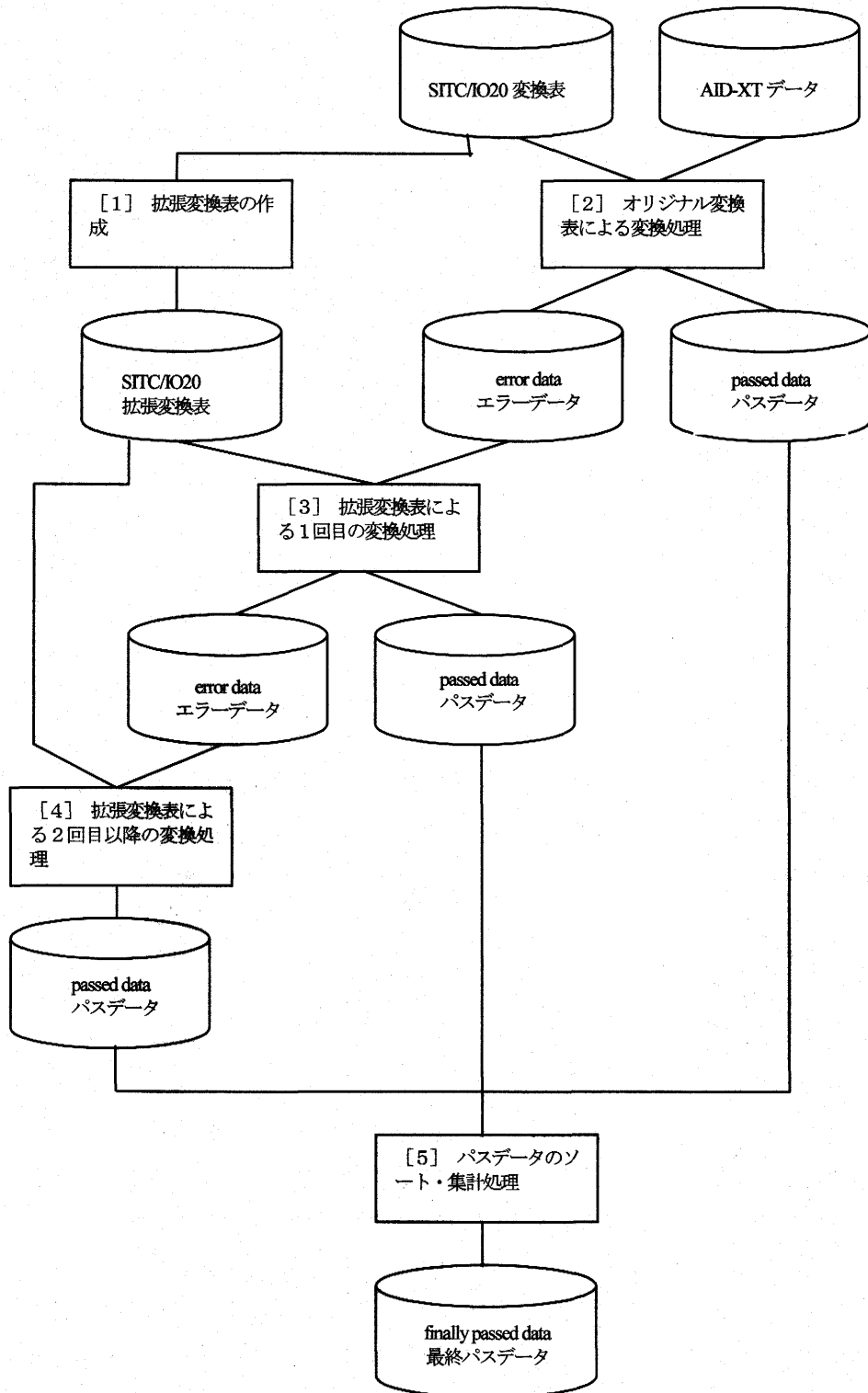


図 10 金額加重配分方式の流れ図

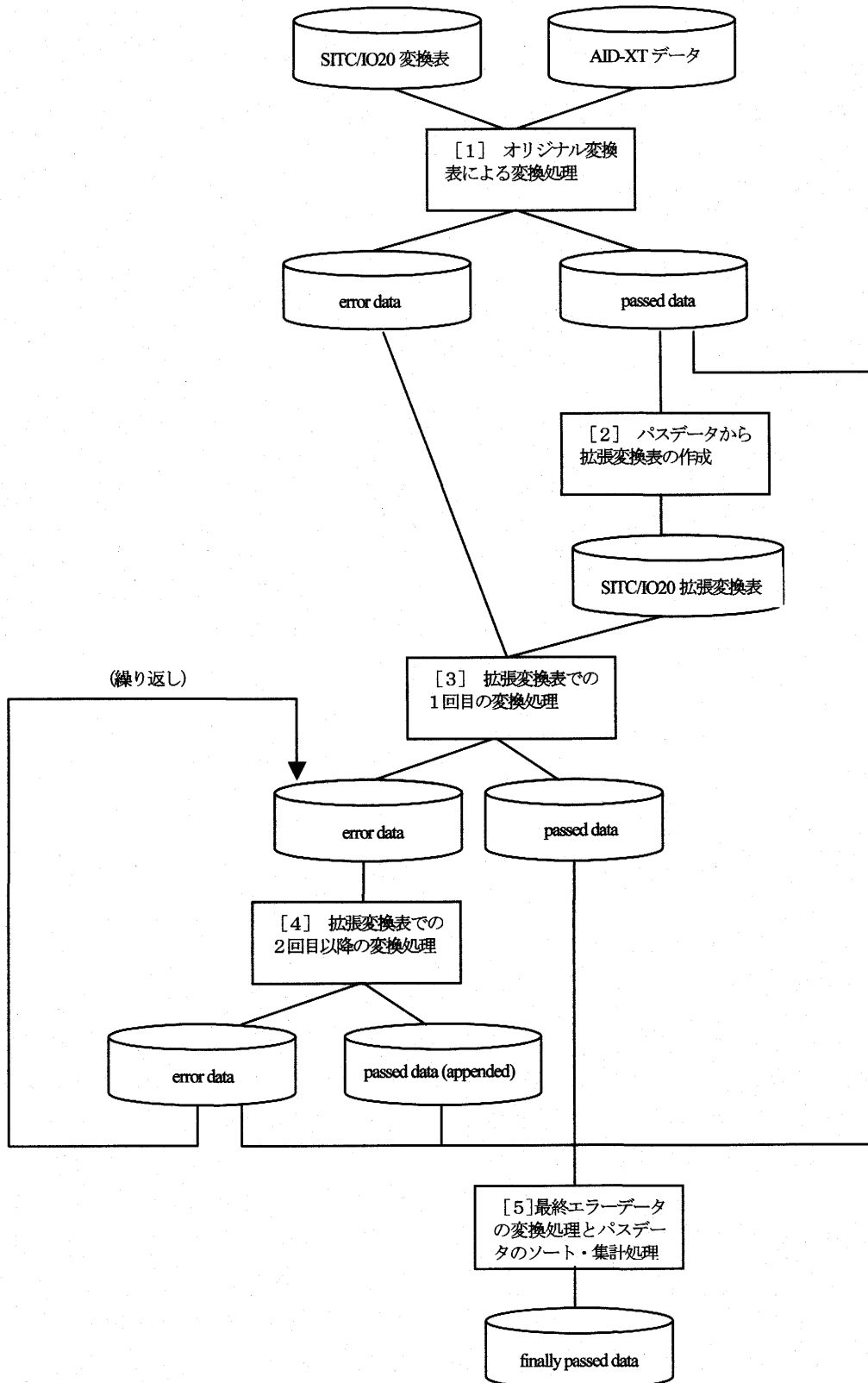


表1 均等配分による変換の実行経過

(金額の単位 100 万 US\$)

	SITC-R1		SITC-R2		SITC-R3	
	件数	金額	件数	金額	件数	金額
対応表	1,309	-	1,832	-	2,667	-
[1] 拡張対応表						
	1,822	-	2,570	-	4,041	-
[2] 対応表による変換						
入力データ	409,297	3,608,473	158,541	5,114,987	326,707	8,771,994
パスデータ	391,041	3,514,793	158,165	5,091,138	240,480	4,903,506
(%)	(95.5)	(97.4)	(99.8)	(99.5)	(73.6)	(55.9)
エラーデー	18,256	93,680	376	23,850	86,227	3,868,489
タ (%)	(4.5)	(2.6)	(0.2)	(0.5)	(26.4)	(44.1)
[3] 拡張対応表による変換 (1回)						
入力データ	18,259	93,680	376	23,850	86,277	3,868,489
パスデータ	15,122	80,844	344	7,750	30,784	2,299,029
(%)	-	(86.3)	-	(32.5)	-	(59.4)
エラーデー	3,875	12,836	32	16,100	55,443	1,569,459
タ (%)	(21.2)	(13.7)	(8.5)	(67.5)	(64.3)	(40.6)
[4] 拡張対応表による変換 (2~n回)						
入力データ	3,875	12,836	32	16,100	55,443	1,569,459
パスデータ	5,300	12,836	34	16,100	55,443	1,569,459
(%)	-	(100.0)	-	(100.0)	-	(100.0)
エラーデー	0	0	0	0	0	0
タ (%)	(0.0)	(0.0)	(0.0)	(0.0)	(0.0)	(0.0)

(出所) 筆者作成

(注) [] の数字は均等配分方式の処理手順 (1.1) の説明で使われている番号を示す。[4] の (2~n回) は、2回目から n 回目までの変換処理をした結果を示す。

チング処理であるため、AID-XT 基礎データの SITC の末尾を 1 桁減らすプログラム、ソート集計、マッチングのプログラムが一つの処理の固まりとなっており、この処理をパスデータが出なくなるまで繰り返すということになる。図9は均等配分方式の流れ図であり、図10は金額加重配分方式の流れ図である。図10に繰り返しの部分の存在を確認できる。桁数が最長の 5 桁の SITC の場合を想定した場合、最悪でも [4] の繰り返しが 4 回繰り返せば 1 桁になる。

[5] 最終エラーデータの変換処理とパスデータのソート・集計処理：均等配分と異なり、金額加重配分では SITC を 1 桁レベルにしても最後まで変換出来ない AID-XT 基礎データがあり

うる。これは以下のような 2 つの条件がそろった場合に発生する。

1 つめの条件は、[1] のステップでエラーになるということである。もう 1 つの条件は、最終エラーデータと報告国、輸出入区分、年、商品分類 1 桁レベルが等しいデータが [1] のステップで 1 件もパスしないということである。これらの条件が揃うと最終的に変換出来ないエラーデータとなる^(註4)。このエラーデータはこれまでの処理で変換出来ないデータとして、IO24 の「その他」のコード(99)を割り当てる。最後に、最終エラーデータとパスデータとを一本化してソート、集計する。

表2 金額加重配分による変換の実行経過

(金額の単位 100 万 US\$)

	SITC-R1		SITC-R2		SITC-R3	
	件数	金額	件数	金額	件数	金額
対応表	1,309	-	1,832	-	2,667	-
[1] 対応表による変換						
入力データ	409,297	3,608,473	158,541	5,114,987	326,707	8,771,994
パスデータ	391,041	3,514,793	158,165	5,091,138	240,480	4,903,506
(%)	(95.5)	(97.4)	(99.8)	(99.5)	(73.6)	(55.9)
エラーデータ	18,256	93,680	376	23,850	86,227	3,868,489
タ (%)	(4.5)	(2.6)	(0.2)	(0.5)	(26.4)	(44.1)
[2] 拡張対応表	593,271	-	233,135	-	351,876	-
[3] 拡張対応表による変換 (1回め)						
入力データ	18,256	93,680	376	23,850	86,277	3,868,489
パスデータ	0	0	0	0	0	0
(%)	-	(0.0)	-	(0.0)	-	(0.0)
エラーデータ	18,256	93,680	376	23,850	86,277	3,868,489
タ (%)	(100.0)	(100.0)	(100.0)	(100.0)	(100.0)	(100.0)
[4] 拡張対応表による変換 (2回め)						
入力データ	12,646	93,680	304	23,850	40,514	3,868,489
パスデータ	8,126	48,946	229	13,203	23,024	1,500,503
(%)	-	(52.2)	-	(55.4)	-	(38.8)
エラーデータ	5,095	44,734	85	10,647	18,654	2,367,986
タ (%)	(40.3)	(47.8)	(28.0)	(44.6)	(46.0)	(61.2)
[4] 拡張対応表による変換 (3回め)						
入力データ	3,477	44,734	59	10,647	14,186	2,367,986
パスデータ	3,910	40,100	76	9,924	14,390	1,951,765
(%)	-	(89.6)	-	(93.2)	-	(82.4)
エラーデータ	793	4,634	6	723	2,242	416,220
タ (%)	(22.8)	(10.4)	(10.2)	(6.8)	(15.8)	(17.6)
[4] 拡張対応表による変換 (4回め)						
入力データ	604	4,634	6	723	1,883	416,220
パスデータ	1,511	4,623	9	723	3,055	355,061
(%)	-	(99.8)	-	(100.0)	-	(85.3)
エラーデータ	55	11	0	0	147	61,159
タ (%)	(9.1)	(0.2)	(0.0)	(0.0)	(7.8)	(14.7)
[4] 拡張対応表による変換 (5回め)						
入力データ	55	11	0	0	147	61,159
パスデータ	26	3	0	0	390	12,938
(%)	-	(31.0)	-	(0.0)	-	(21.2)
エラーデータ	43	8	0	0	49	48,221
タ (%)	(78.2)	(69.0)	(0.0)	(0.0)	(33.3)	(78.8)

(出所) 筆者作成

(注) [] の中の数字は金額加重配分方式の処理手順 (2.1) の説明で使われている番号を示す。入力データの内訳としてパスデータとエラーデータがありその比率が () の中に%で示される。

2.2 特徴

金額加重配分方式の長所は、金額実績に基づいて配分するため、均等配分によるものに比べて配分の妥当性が向上する可能性があるという点にある。ただしこれは前述の手順 [1] でパステータが多く、[2] で有効な拡張対応表が作成できた場合の話である。

逆に短所としては、前述のように、最終的にエラーデータが無くならないことがありうることである。手順 [1] でパステータが少なかつたような場合は [2] で多くの種類のキーを含む有効な拡張対応表が作成できないため、最終エラーデータが多くなってしまう。このような場合は一概に均等配分に比べ妥当性が高いとは断言できない。金額加重配分方式の妥当性はひとえにオリジナルの対応表の完成度にかかっているとみえる。

3. 変換結果と評価

ここでは、実際に前述の2つの変換法を使って変換処理を行った結果をまとめ、それについて若干の評価をしたい。

まず、今回の処理に使用されたデータの条件を明らかにしておく。報告国が以下の13の国・地域である。日本、中国、香港、台湾、韓国、マレーシア、シンガポール、ヴィエトナム、インドネシア、フィリピン、タイ、インド、パキスタン。報告年は国・地域によって異なるが1962年から1993年の間である^(註5)。

均等配分による変換を行った結果が表1である。この表は処理の各段階で入力データ全体のうちパステータとエラーデータが占める比率を示したものである。例えばSITC-R1の場合、手順[2]で当初の入力データの件数にして95.5%が変換され、4.5%がエラーとなっている。次の手順[3]で配分処理が行われ、この段階でエ

ラーは21.2%出力されている。なお、パステータの件数%欄は、全て「-」となっている。パステータは入力データの1件に対して複数件出力されている可能性があるため、パステータ件数を入力データ件数で単純に割って比率を出しても意味がないためである。

SITC各改訂版を横並びに見ると手順[2]での変換エラーの割合がSITC-R1とSITC-R2が件数比でも金額比でも数%どまりなのにSITC-R3は件数比で26.4%、金額比で44.1%と大きな値となっている。また、手順[3]が終わった段階でのエラー比率を金額比で出すと、SITC-R1は $12836/3608473=0.4\%$ 、SITC-R2は、0.3%、SITC-R3は17.9%となっており、やはりSITC-R3が突出していることがわかる。手順[3]だけでのエラーの比率は手順[2]の比率のように一様な傾向はみられないため、手順[2]でのエラー比率が高かったこと、つまりオリジナルの対応表の完成度が手順[3]終了後のエラー比率に影響しているといえる。

金額加重配分方式による結果が表2である。金額加重配分方式では、手順[1]でオリジナルの対応表で変換を行うところは均等配分方式と全く同様である。一方、手順[4]での各段階が手動で繰り返し処理されているので、各処理比率がわかるようになっている。

やはり均等配分方式でみたように、SITC-R3のエラー比率がSITC-R1、SITC-R2に比べて全体に高くなっている。また[4]拡張対応表による変換(5)が終了した最終時点でのエラー比率(金額比)を出す、SITC-R1で0.0002%、SITC-R2で0%なのに、SITC-R3では0.5497%となっている。均等配分方式と同様、オリジナルの対応表の影響が現れているといえる。

おわりに

これまで見てきたように、エラーデータの処

理において、均等配分、金額加重配分のいずれの方式を採った場合でもオリジナルの対応表の完成度が重要である。なぜなら、均等配分方式ではエラーデータはオリジナルの対応表から作成された拡張対応表を元に処理されるし、金額加重配分方式でもオリジナルの対応表によって変換できたパスデータを元に拡張対応表を作成してエラー処理を行うからである。

さて、これら二つの変換方式をどのように使い分けたらよいかという問題には今回は言及しなかった。言い換えれば、どのような場合に各変換方式の妥当性が高まるのか、また、妥当性の数値化ができるかという問題である。このような二つの方式の比較検討は今後の課題である。

(注1) なお、以下で挙げる例における金額の値はあくまでもサンプルであり、実際の値ではないことをお断りしておく。

(注2) 24+1=25回の配列ではなく、20+1=21回の配列に展開する理由は、IO24のうち実際に使用されるのが01から20までの商品貿易に関わる分類だけであるためである。

(注3) これは同時に処理する報告国数にもよる。今回行った14ヶ国の場合、データ件数は593271、ファイル容量は91MB(SITC-R1)である。

(注4) 具体例を挙げる。最終エラーデータには図13のようなデータが存在する。(金額Vは実際のデータの値とは異なる)

図13 最終エラーデータ

RC	DT	Y	SITC	V
117310	2	90	9	102

これと同じキーを持ち、手順[1]で出力されたエラーデータを確認すると、図14のようになっている。

図14 手順[1]でのエラーデータ

RC	DT	Y	SITC	V
117310	2	90	9310	100
117310	2	90	9610	2

図15のように、オリジナル対応表では、SITCの9310、9610のデータは存在しない。そのため1番目の条件

に合致する。また手順[1]で、図16のようにエラーデータとSITCの1桁レベルまでのキーが同じデータが1件でもパスデータになっているかどうかを確認すると、存在しない。そのため拡張対応表にもこのキーを持つデータは作成されずそのためにエラーデータが残ったことがわかる。これは2番目の条件に合致する。

図15 SITC/IO24 対応表

SITC	IO24
91107	19
93107	19
96107	19
97101	19
:	

図16 キーがパスデータに存在しない

RC	DT	Y	SITC	V
117310	2	90	9	

付け加えると、図15の対応表には、SITCの93107、96107を持つデータがある。均等配分方式であれば、これらのデータは手順[2]で拡張対応表にSITCの9310、9610に該当するデータができておりエラーデータはすんなりパスしてははずである。金額加重配分方式では拡張対応表にSITC以外のキーを持つため、パスデータに図16のようなキーを持つデータが無いこの例のような場合、最後まで変換出来ない結果となってしまう。

(注5) ここで使用した入力データは、SITC4桁、5桁の整合性を「補正」する前のAID-XT基礎データである。一方、本書の第2部の表、グラフは、「補正」後のAID-XT基礎データを入力データとしていることに注意されたい。「補正」の概念については、本書第2章「世界貿易マトリクス作成における整合性評価」を参照されたい。

【参考文献】

- [1] 古河俊一・野田容助共著『標準国際貿易商品分類と産業分類の対応関係』統計資料シリーズNo.80 アジア経済研究所 1998
- [2] 黒子正人「商品分類の産業分類への変換—変換エラーデータ処理のプログラム—」(野田容助編『世界貿易マトリクスの作成と評価—貿易指数の推計に向けて—』調査研究報告書 アジア経済研究所 2002)