Product-to-Product Tables via Product-Technology with No Negative Flows

CLOPPER ALMON
ABSTRACT *Two ways of making a product-to-product table are commonly recognized, one based on the product-technology assumption and the other on the industry-technology assumption. The second is recognized as highly implausible but is often used because the first frequently leads to small negative flows which make no economic sense. This paper shows how a slight adjustment in the product-technology assumption leads to an algorithm that is certain to avoid negative flows yet keep close to the spirit of the product- technology assumption. Some details of the application of this method to the USA table for 1992 are reported. Similar applications to every American table since 1958 have given consistently sensible results. A computer program for the method is included.*

KEYWORDS: *Product-to-product input-output tables; product technology; symmetric input-output tables, industry technology*

Clopper Almon, Department of Economics, University of Maryland, College Park, MD 20740 USA.
Email: Almon@inforum.umd.edu

1. The Problem

Makers of input-output tables often find data on inputs not by the *product* into which they went but by the *industry* that used them. An *industry* is a collection of establishments with a common principal product. But besides this principal product, any one of these establishments may produce a number of secondary products, products primary to other industries. Establishments classified in the Cheese industry may also produce ice cream, fluid milk, or even plastic moldings. Consequently, the Cheese industry may have inputs of chocolate, strawberries, sugar, plastic resins, and other ingredients that would appal a connoisseur of cheese. The inputs, however, are designated by what the product was, not by what industry made them. Similarly, data on the final demands, such as exports and personal consumption expenditure, is by product exported or consumed, not by the industry which made it. Thus, input-output matrices usually appear in two parts. The first part, called the Use matrix, has products in its rows but industries in its columns. The entries show the use of each product (in the rows) by each industry (in the columns.) The second, called the Make matrix, has industries in the rows and products in the columns; the entries show how much of each product was made in each industry.

How can we use these two matrices to compute the outputs of the various products necessary to meet a final demand given in product terms?

One way is to consider that each product will be produced in the various industries in the same proportion as in the base year of the table. This assumption is used, for example, in computable general equilibrium models based on social accounting matrices that explicitly show the Make and Use matrices. This assumption, however, produces anomalous — not to say silly — results. In the above example, an increase in the demand for cheese would automatically and immediately increase demand for chocolate, strawberries, and sugar. That is nonsense. There must be a better way to handle the problem.

This highly unsatisfactory situation has led to efforts to make a product-to-product matrix. To make such a matrix, we need to employ an additional assumption. There are basically two alternatives:

1. The product-technology assumption, which supposes that a given product is made with the same inputs no matter which industry it is made in.
2. The industry-technology assumption, which supposes that all products made within an industry are made with the same mix of inputs.

The *System of National Accounts 1993* (SNA) reviews the two assumptions and finds (Section 15.146, p. 367) "On theoretical grounds, ... the industry technology assumption performs rather poorly" and is "highly implausible." (Section 15.146, p 367) "From the same theoretical point of view, the product (commodity) technology model seems to meet the most desirable properties .... It also appeals to common sense as it is found *a priori* more plausible than the industry technology assumption. While the product technology assumption thus is favoured from a theoretical and common sense viewpoint, it may need some kind of adjustment in practice. The automatic application of this method has often shown results that are unacceptable, insofar as the input-output coefficients appear as extremely improbable or even impossible. There are numerous examples of the method leading to negative coefficients which are clearly nonsensical from an economic point of view." (Section 15.147)

Since about 1967, the Inforum group has used an "semi-automatic" method of making "some kind of adjustment" in calculations based on the product-technology assumption, as called for by the SNA. We have used it with satisfactory results -- and without a single negative coefficient -- on every American table since 1958. The method was published in Almon *et al.* in 1974. Despite this long and satisfactory use of the method, it seems not to have come to the attention of the general input-output community. In particular, the authors of the section quoted from the SNA seem to have been unaware of it. The purpose of this note is to record the method where it is more likely to come to the attention of anyone working in input-output. At the same time, it expands the previous exposition with an example, provides a computer program in the C++ language for executing the method, and presents some of the experience of applying the method to the 1992 table for the USA.

2. An Example

An example will help us to visualize the problem. The Table 1 below shows the Use matrix for a 5-sector economy with a strong concentration in dairy products, especially cheese and ice cream.

Table 1. The Use Matrix

USE	Industries				
Products	Cheese	Ice cream	Chocolate	Rennet	Other
Cheese	0	0	0	0	0
Ice cream	0	0	0	0	0
Chocolate	4	36	0	0	0
Rennet	14	6	0	0	0
Other	28	72	30	5	0

We will call this matrix U. The use of chocolate in makings cheese and rennet in making ice cream alerts us to the fact that the columns are industries, not products. (Rennet is a substance used to make milk curdle. It is commonly used in making cheese but never in ice cream.) The Make matrix, shown in Table 2 below, confirms that cheese is being made in the ice cream industry and ice cream in the cheese industry.

Table 2. The Make Matrix

MAKE	Products				
Industries	Cheese	Ice cream	Chocolate	Rennet	Other
Cheese	70	20	0	0	0
Ice cream	30	180	0	0	0
Chocolate	0	0	100	0	0
Rennet	0	0	0	20	0
Other	0	0	0	0	535
Total	100	200	100	20	535

This matrix shows that of the total output of 100 of cheese, 70 was made in the Cheese industry and 30 in the Ice cream industry, while of the total ice cream output of 200, 180 was in the Ice cream industry and 20 in the Cheese industry. It also shows that, of the total output of 90 by the cheese industry, 78 percent (70/90 = .77778) was cheese and 12 percent ice cream. We will need the matrix, M, derived from the Make matrix by dividing each cell by the column total. For our example, the M matrix is shown in Table 3.

Table 3. The M Matrix

M	Cheese	Ice cream	Chocolate	Rennet	Other
Cheese	0.7	0.1	0.0	0.0	0.0
Ice cream	0.3	0.9	0.0	0.0	0.0
Chocolate	0.0	0.0	1.0	0.0	0.0
Rennet	0.0	0.0	0.0	1.0	0.0
Other	0.0	0.0	0.0	0.0	1.0

Now let us suppose that, in fact, cheese is made by the same recipe wherever it is made and ice cream likewise. That is, we will make the ?product-technology assumption.” If it is true and the matrices made well, then there exists a “recipe” matrix, R, in which the first column shows the inputs into cheese regardless of where it is made, the second column shows the inputs into ice cream regardless of where it is made, and so on. Now the first column of U, U_1, must be $.70*R_1 + .10*R_2$,where R_1 and R_2 are the first and second columns of R, respectively. Why? Because the Cheese plants make 70 percent of the cheese and ten percent of the ice cream. In general,

$$U = RM' \qquad (1)$$

where M' is the transpose of M. It is then a simple matter to compute R as

$$R = U(M')^{-1}.$$

For our example, $(M')^{-1}$ is given in Table 4.

Table 4. M' Inverse

1.5	-0.5	0.0	0.0	0.0
-0.2	1.2	0.0	0.0	0.0
0.0	0.0	1.0	0.0	0.0
0.0	0.0	0.0	1.0	0.0
0.0	0.0	0.0	0.0	1.0

and R works out to be

Table 5. The R or "Recipe" Matrix

R	Cheese	Ice cream	Chocolate	Rennet	Other
Cheese	0	0	0	0	0
Ice cream	0	0	0	0	0
Chocolate	0	40	0	0	0
Rennet	20	0	0	0	0
Other	30	70	30	5	0

This R is very neat. All the rennet goes into cheese and all the chocolate goes into ice cream. Unfortunately, as indicated by the quotation from the SNA, it is rare for the results to turn out so nicely.

Indeed, just a slight change in the U matrix will show us what generally happens. Suppose that the U matrix had been just slightly different, with 1 unit less of chocolate going into cheese as shown below and one less unit of rennet used in ice cream.

Table 6. An Alternative Use Matrix

Alternative U	Cheese	Ice cream	Chocolate	Rennet	Other
Cheese	0	0	0	0	0
Ice cream	0	0	0	0	0
Chocolate	3	37	0	0	0
Rennet	15	5	0	0	0
Other	28	72	30	5	0

Table 7 shows what the R matrix would have been:

Table 7. An Impossible R Matrix

Impossible R	Cheese	Ice cream	Chocolate	Rennet	Other
Cheese	0.0	0.0	0.0	0.0	0.0
Ice cream	0.0	0.0	0.0	0.0	0.0
Chocolate	-1.7	41.7	0.0	0.0	0.0
Rennet	21.7	-1.7	0.0	0.0	0.0
Other	30.0	70.0	30.0	5.0	0.0

Here we find the infamous small negative flows. It is not hard to see how they arise. While it is conceivable that the Cheese industry does not produce chocolate ice cream, it is also very easy for the table makers to forget to put into the Cheese industry the chocolate necessary for the ice cream it produces, or to put in too little. Wherever that happens, negatives will show up in the R matrix.

The negatives have driven at least some statistical offices to the industry-technology assumption. The so-called commodity-to-commodity matrix, C, derived from this assumption is

$$C = UN', \tag{3}$$

where N is the matrix derived from the Make matrix by dividing each row by the row total. For example, the Cheese column of C is $C_1 = .77778U_1 + 0.14285U_2$ because 77.778 percent of the product of the first industry is cheese and 14.285 percent of the product of the second industry is cheese. The result of applying this assumption to our example is Table 8.

Table 8. The Mess Made by the Industry Technology Assumption

C Indust. Tech.	Cheese	Ice cream	Chocolate	Rennet	Other
Cheese	0	0	0	0	0
Ice cream	0	0	0	0	0
Chocolate	8.254	31.746	0	0	0
Rennet	11.746	8.254	0	0	0
Other	32.063	67.936	30	5	0

This “solution” has made matters worse. The original U matrix had 4 units of chocolate going into the Cheese industry, which admittedly made some ice cream. Now this industry-technology product-to-product matrix asserts that *8.25 units of chocolate went into producing pure cheese*! Not into the Cheese *industry* but into the *product* cheese! And 8.25 units of rennet went into producing curdled ice cream! To call the result a product-to-product table would be little short of scandalous.

Fortunately, we do not have to choose between this sort of massive nonsense and negative flows. It is perfectly easy to rely mainly on the product-technology assumption, yet avoid the negatives, as we will now show.

(The idea to compute R from equation (1) seems to have been first put in print by Van Rijckeghem (1967). He realized that there could be negatives but did not think they would be a serious problem. The idea of using equation (1) in this way, however, must have been in the air, for by early 1967, I had used it, without thinking that it was original, found negatives, and started work on the algorithm presented below.)

3. The No-Nonsense Product-Technology Algorithm

We wrote the basic equation relating U, M, and R as equation (1) above. It will prove convenient to rewrite equation (1) as

$$U' = MR'. \qquad (4)$$

Using U'_i to denote the i^{th} column of U' and R'_i to denote the i^{tb} column of R, we can write

$$U'_i = MR'_i \, . \qquad (5)$$

Notice that this is an equation for the distribution of product i in row i of the Use matrix as a function of M and distribution of the same product in row i of the R matrix. We can simplify the notation by writing

$$u = U'_i \quad \text{and} \quad r = R'_i \, ; \qquad (6)$$

then the previous equation becomes

$$u = Mr \qquad (7)$$

or

$$0 = -Mr + u \qquad (8)$$

and adding r to both sides gives

$$r = (I - M)r + u \, . \qquad (9)$$

Save in the unusual case in which less than half of the production of a product is in its primary industry, the column sums of the absolute values of the elements of (I - M) are less than 1, and the convergence of the Seidel iterative process for solving this equation is guaranteed a by well-known theorem. (If the share of the total production of a particular product coming from the industry to which it is primary is x, then the absolute value of the diagonal of (I - M) for that product is $| 1 - x |$ and the sum of all the absolute values of off-diagonal elements in the column is $|1 - x |$, so the total for the column is $2*| 1 - x |$, which is less than 1 if $x > .5$.) We start this process with

$$r^{(0)} = u \qquad (10)$$

and then define successive approximations by

$$r^{(k+1)} = (I - M) \, r^{(k)} + u \, . \qquad (11)$$

To see the economic interpretation of this equation, let us write out the equation for the use of a product, say chocolate, in producing product j, say cheese:

$$r_j^{(k+1)} = u_j - \sum_{\substack{h=1 \\ h \neq j}}^{n} m_{jh} r_h^{(k)} + (1 - m_{jj}) r_j^{(k)} \qquad (12)$$

The first term on the right tells us to begin with the chocolate purchases by the establishments in the cheese industry. The second term directs us to remove the amounts of chocolate needed for making the secondary products of those establishments by using our present estimate of the technology used for making those products, $r^{(k)}$. Finally, the last term causes us to add back the chocolate used in making cheese in other industries. The amount of chocolate added by the third term is exactly equal to the amount stolen, via second terms, from other industries on account of their production of product j:

$$(1 - m_{jj}) r_j^{(k)} = \sum_{\substack{h=1 \\ h \neq j}}^{n} m_{hj} r_j^{(k)} \qquad (13)$$

because

$$\sum_{h=1}^{n} m_{hj} = 1. \qquad (14)$$

It is now clear how to keep the negative elements out of *r*. When the “removal” term, the second on the right of (12), is larger than the entry in the Use matrix from which it is being removed, we just scale down all components of the removal term to leave a zero balance. Then instead of adding back the “total-stolen-from-other-industries” term, $(1 - m_{jj})r_j$, all at once, we add it back bit-by-bit as it is captured. If a plundered industry, say Cheese, runs out of chocolate with only half of the total chocolate claims on it satisfied, we simply add only half of each plundering product's claim into that product's chocolate cell in the R matrix. We will call the situation where the plundered industry runs out of the product being removed before all claims are satisfied a “stop”.

The process can also be applied to the rows of the value added part of the matrix. It is not certain, however, that the column sums of the resulting value-added table will match the value added as calculated from product output minus intermediate input. This value-added matrix will generally require RAS balancing to make it consistent with the product-to-product intermediate table.

4. When is it appropriate to use this algorithm?

This algorithm is appropriate where the product-technology assumption itself is at least approximately true. Essentially, it allows there to have been slightly different technologies in industries where assuming *strictly* the average product technology would produce negatives. It is appropriate where the negatives arise because of inexactness in making the tables or because of slight differences in technologies in different industries. Applied to the Use matrix of either Table 1 or Table 6, this method gives the ?neat" Recipe matrix of Table 5 with no rennet in ice cream and no chocolate in cheese. It never produces negative entries nor positive entries where Use has a zero. The row totals are unaffected by the process. It is, moreover, equivalent to deriving Recipe from equation (1) if no negatives would arise, so that if the product-technology assumption is strictly consistent with the Use and Make tables, the method produces the true matrix. It may even produce a correct Recipe matrix from a faulty Use matrix — as it has perhaps done in our example — so that equation (1) could be used to revise the estimate of the Use matrix.

Certain accounting practices, however, may produce situations which appear to be incompatible with the product-technology assumption, even though the underlying reality is quite compatible. For example, local electric utilities generally buy electricity and distribute it. In the U.S. tables, they are shown as buying electricity (not coal), adding a few intermediate inputs and labor, and producing only a secondary product, electricity, which is transferred, via the Make matrix, back to electricity. Looked at mechanically, this method of making electricity is radically different from that used in the Electricity industry, which uses coal, oil, and gas to make electricity, not electricity itself. If our algorithm is applied thoughtlessly to this situation, it cannot be expected to give very sensible results.

Fortunately, it is easy to generate signs of this sort of problem. One can compute the new Use matrix implied by equation (1) with the Recipe matrix found by the algorithm and the given Make matrix. This "NewUse" matrix can then be compared with the original Use matrix and the causes of the differences investigated. We will follow this procedure in next section on the experience of using the method on the 1992 tables for the USA.

To fix the problem in the above example about electricity, we have only to consider the output of the State and local utilities as production of their own primary product, which is then sold, via the Use matrix — not transferred via the Make matrix — to the Electricity industry.

Thus, in the use of this method, a number of iterations may be necessary. Changes in concepts, in treatments of some transactions, and occasionally in underlying data may be necessary. Thus, although the calculation of the non-negative Recipe matrix is totally automatic, it may be necessary to make several runs of the program with intermediate adjustments of data before acceptable results are obtained.

In this process, it must be recognized that a nice, clean accounting system may not be operational, that is, it may not provide by itself a simple, automatic way to go from final demand vectors specified by products to total outputs of those products. We may have to change slightly some of the concepts in the accounting system to make it operational. In making the change required for the Electricity example, we have messed up the neat accounting concept of the Electricity column of the Use matrix as a picture of what came into a particular group of establishments. We have, however, taken a step toward creating what might be called an operational Use matrix. I do not say, therefore, that statistical offices should not produce pure accounting Use matrices. But I do feel that they should also prepare the operational use matrix and the final product-to-product matrix, for in the process, they will learn about and deal with the problems which the users of the matrix will certainly encounter. They may even discover and correct errors in their work before they are discovered by their users.

This process is totally inappropriate for handling by-products such as hides produced in the meat packing industry or metal scrap produced in machinery industries. Their treatment is a different subject.

Rainer and Richter [1992] have documented a number of steps which they took towards making what we have called here the operational Use and Make matrices. Such steps should certainly be considered and applied if need. These authors still ended up with hundreds of negative flows in the R matrix because they were using just equation (1). At that point, the process described here could have been applied rather than simply tolerating the negatives.

Steenge and Konijin [1992] point out that if the R matrix computed from equation (1) has any negatives in it, then it is possible to change the levels of output of the various industries in such a way that more of all products is produced *without* using more of all inputs. They feel that it is implausible that such a rearrangement is possible and observe that perhaps the negatives "should not be regarded as rejecting the commodity technology assumption, but as indicators of flaws in the make and use tables." (p. 130). I feel that there is much merit in that comment. It seems to me that the right time and place to use the algorithm presented here is in the process of making the tables. If there are not good statistical grounds for preferring the original Use matrix, the recomputed NewUse might well be argued -- following the reasoning of Steenge and Konijn -- to be a better estimate.

The caveat here is that there may well be cases where it really would be possible to increase the outputs of all products while using less of some product. For example, if there are shoes made in the Plastics products industry without any use of leather, while the Footwear industry uses leather, then by moving shoe production from Footwear to Plastic products it may be possible to produce more of all products while using less leather. Where such cases arise, a different solution is necessary, for example, moving the shoes made in the Plastics products industry together with their inputs into the Footware industry or insisting that the two kinds of shoes are separate if substitutable products.

Finally, the ingenious attempt of ten Raa [1988] to modify elements of the matrices in such a way as to find a most probable U matrix consistent with a non-negative R should be mentioned even though it ended, in the author's view, in frustration. Unfortunately, this otherwise interesting article contains a totally incorrect statement of the method described here and a consequent dismissal of it as "arbitrary."

5. Application to the U.S.A Tables for 1992

The method described here has been applied to all of the USA tables since 1958 with experiences broadly similar to those described here for the 1992 table. This table has 534 sectors, counting some construction sectors which have no intermediate sales. Of these 534, 425 have secondary production. Of the 283,156 possible cells in a 534 X 534 matrix, the Use matrix has 44,900 non-zero cells, and the Make matrix has 5,885. The matrix was produced in two versions. In one, certain activities, such as restaurant services of hotels, were removed from the industry where they were produced (Hotels) and put into the sector where these activities were primary (Restaurants). In the other, these activities were left in the industry where they were conducted. The first version was designed to make the product-technology assumption more valid, and it has been used here. The matrix also puts true by-products (such as hides from meat packing) in a separate row, not one of the 534 considered here.

We will look at the process midway along. That is, some adjustments in the Use and Make matrix from which the algorithm starts will have already been made. As a result of this application, further adjustments will be suggested before the next application.

Before this application of the algorithm, the output of industries which had only secondary production was changed to be primary and the flows moved from the Make to the Use matrix. (This treatment makes good economic sense. It implies that, of a dollar spent on local public transit, a fixed proportion will go to services owned by state and local and local governments. In this case, this assumption does not produce the sort of distortion caused by the mechanically similar assumption that a fixed proportion of all cheese is produced with the input structure of the ice-cream industry.)

In the following rather detailed descriptions, necessary to give a picture of what the process is really like, I will, to avoid confusion, capitalize the first letter of the first word in industry names but not in product names.

The industry Water and sewer systems failed to satisfy the requirement that at least half of the output of a product should be in the industry where it is primary. Indeed, some 85 percent of this product's output comes from Other state and local enterprises, and the iterative procedure failed to converge for a few rows until this secondary transfer was converted into a primary sale. Production of secondary advertising services, which occurred in many sectors, was converted to a primary product of the producing industry and "sold" via the Use matrix to the Advertising industry. Secondary production of recreational services in agricultural industries was similarly converted. Much of the output of the several knitting industries had been treated originally as secondary production, and these had been changed to primary sales before the calculations shown here. Finally, the diagonals of many columns of the Use matrix are large, in part because intra-firm services, such as those of the central offices, often appear there. Thus the same sort of service that is on the diagonal of industry i is also on the diagonal of industry j. In this case, the product-technology assumption does not apply, not because it is untrue, but because of the way the table was made. Until we are able to obtain tables without this problem, we have just removed half of the diagonals from the Use table before calculating Recipe, and have then put back this amount in both of these matrices and in the NewUse matrix.

The data in both Use and Make tables were given to the nearest 1 million dollars, and all dollar figures cited here are in millions. The convergence test in the iterative process was set at one tenth of that amount, .1 million dollars. The iterative process converged for most rows of the R matrix in less than five iterations. The most iterations required for any row was 15.

The resulting Recipe matrix looks very similar in most cells to the original Use table. The Recipe matrix contains, of course, only non-negative entries and can have strictly positive entries only where U has positive entries. It may, however, as a result of the "robbing" process, have a zero where U has a positive entry. In all, there were only 95 cells in which Recipe had a zero where Use had a positive entry.

Although it is the Recipe matrix that we need from this process, it is also interesting, as noted above, to compare the original Use matrix with what we may call NewUse, computed according to equation 1 by NewUse = Recipe*Make'. The difference between Use and NewUse shows the changes in the Use matrix necessary to make it strictly compatible with product-technology assumption, the given Make matrix, and the calculated Recipe matrix. If there was no "stop" in a row, the two matrices will be identical in that row. There were 118 such identical rows, 109 of them having no secondary output.

In the other rows, these differences turn out to be mostly small but very numerous. The first and most striking difference is that NewUse has almost twice as many non-zero cells as does Use. Nearly all of these extra non-zeros are very small, exactly the sort of thing to be reasonably ignored in the process of making a table. But it is precisely this "reasonable ignoring" that leads to the problem of many small negatives in the product-to-product tables calculated without the "no-nonsese" algorithm.

To get a closer look at how Use and NewUse compare, we may first divide each column by corresponding industry's output and then look at the column sums of the absolute values of the differences of individual coefficients in the column. This comparison is shown in Table 9. Clearly the vast majority of industries show only small differences compatible with "reasonable ignoring" of small flows in the Use matrix. They, therefore, cast no serious doubt on the product-technology assumption or the usability of the Recipe matrix obtained by the no-nonsense algorithm. If what we are interested in is the R matrix, we can ignore the small differences between Use and NewUse.

Table 9. Comparison of Use and NewUse

Sum of Absolute Differences	Count
.050 - .225	17
.030 - .050	24
.020 - .030	54
.010 - .020	117
.000 - .010	312

Table 10. Largest Differences between Use and NewUse

Sum |dif|	Column numb.	Column Name	Largest single difference Row	| dif |	Row name
0.250	272	Asbestos products	31	0.023	Misc. nonmetallic minerals
0.232	88	Sausages	3	0.151	Meat animals
0.167	125	Vegetable oil mills, nec	15	0.074	Oil bearing crops incl s
0.118	493	Auto rental & leasing	232	0.025	Petroleum refining
0.088	128	Edible fats and oils, nec	15	0.043	Oil bearing crops incl s
0.088	126	Animal & marine fats	126	0.038	Animal & marine fats &
0.086	87	Meat packing plants	3	0.057	Meat animals
0.079	285	Primary metals, nec	22	0.006	Iron & ferroalloy ore m
0.079	225	Manmade organic fibers	212	0.036	Indl chem: inorg & org
0.074	450	Transportation services	232	0.019	Petroleum refining
0.068	123	Cottonseed oil mills	5	0.048	Cotton
0.065	357	Carburetors, pistons,	391	0.011	Electronic components
0.060	99	Pickles, sauces	1	0.011	Dairy farm products
0.060	95	Canned & cured sea food	19	0.039	Commercial fishing
0.059	139	Yarn mills & textile fini	212	0.035	Indl chem: inorg & org
0.055	459	Sanitary services, steam	413	0.018	Mechanical measuring devices
0.051	248	Leather gloves	244	0.012	Leather tanning

There are, however, a few cases that should be looked at more closely. Table 10 shows a list of all of industries which had a sum of absolute differences greater than .050. We will look at the top five.

For Asbestos products, the cause of the difference is quickly found. The fundamental raw material for these products comes from industry 31 Misc. non-metallic minerals. Over forty percent of the output of asbestos products, however, is produced in industry 400 Motor vehicle parts and accessories, but this industry buys neither miscellaneous non-metallic minerals nor asbestos products. In other words, it seems to be making almost half of the asbestos products without any visible source of asbestos. This anomaly seems to me to be an oversight in the Use matrix which should be simply corrected. If our only interest is the Recipe matrix, the algorithm seems to have computed pretty nearly the right result from the wrong data. On the other hand, if we want to correct the Use table, NewUse, gets us started with the right entry for Misc. non-metallic minerals into both Motor vehicle parts and Asbestos products. To keep the right totals in these two columns of Use will require manual adjustments.

The second largest difference between Use and NewUse shown in Table 10 is in the input of meat animals into Sausage. The Sausage industry is shown in the Use matrix to buy both animals ($655) and slaughtered meat ($9688). It had a primary output of $13458 and a secondary output of $2612 of products primary to Meat packing. Meat packing had a secondary output of $4349 of sausage. Now in Meat packing, the cost of the animals is over eighty percent of the value of the finished product, so the purchases of animals in the Sausage industry is insufficient to cover even the secondary meat output of this industry, not to mention making any sausage. In making Recipe, the input of animals directly into sausage is driven to zero and cut off there rather than being allowed to become negative. Then when NewUse is made, the direct animal input for all the secondary production of meat packing products is put in, thus making a flow some six times as large as the purchase of meat animals by the Sausage industry in the original Use matrix.

What I believe to be really happening here is that Sausage plants are mostly buying slaughtered meat from meat packers, selling off the best cuts as a secondary product, and using the rest to make sausage. Over in the Meat packing plants the same thing is happening. Fundamentally, there is only one process of sausage making. The question is how to represent it in the input-output framework. The simplest representation of it in the Use matrix would be to have packing houses sell to sausage plants only the meat that would be directly used in sausage. The rest, the choice cuts sold off as meat by Sausage mills, would simply be considered sold by the packers without ever passing through the Sausage mills. The industry output of Sausage mills is reduced but cost of materials (namely, meat) is reduced by exactly the same amount, so there is no need to adjust other flows. Product output of meat is reduced, but not the industry output. Thus, a slight adjustment in the accounting makes it broadly compatible with the product-technology assumption. The seventh item in Table 10, by the way, is just the other side of this problem.

The third largest of the discrepancies lies in row 16, oil-bearing crops, of industry 125 Vegetable oil mills n.e.c (not elsewhere classified). The differences in the underlying flows is not large, $298 in Use and $251 in NewUse, but it turns up in Table 10 because the cost of these oil crops is such a large fraction of the output of the Vegetable oil mills. A comparison of the oil-bearing crops row of Use and NewUse shows that NewUse has a number of small positive entries for industries where, as for Cheese, Use has a zero and where, moreover, it is highly implausible that there was any use of oil seeds. On the other hand, most of the large users of oil seeds, like Vegetable oil mills have had their usage trimmed back. The key to what is going on is found in industry 132 Food preparations n.e.c.. In Use, this industry bought $558 from oil bearing crops, nearly twice the consumption of the vegetable oil mills themselves. Salad dressings, it seems, are in this industry. That fact, by itself, is not a problem. The problem is that about a quarter of the production of products primary to this industry are made in other industries. In fact, most of the food manufacturing industries have some secondary production of the miscellaneous food preparations. Probably “preparations” made in the Cheese industry are quite different from those made in the Pickles industry. And it certainly makes no sense to spread oil seed inputs all over the food industries. Here we have a clear case of the inapplicability of the product-technology assumption if all these secondary products are considered to be truly the same product. On the other hand, the very heterogeneity of the products makes it appropriate to consider each as a primary product of the industry which produces it and then “sell” it, via the Use matrix, to Food preparations for distribution. In the next pass at making Recipe, this change is to be made.

The vegetable oil industries also present another interesting case of apparent but perhaps not real violation of the product-technology assumption, which shows up in the fifth item in Table 10. Industry 125 Vegetable oil mills n.e.c. has inputs of oil-bearing crops, cotton, and tree nuts totaling $437. It uses these oil sources to produce a primary output of $572. Industry 128 "Edible fats and oils" produces $92 more of products primary to 125 without a penny of any of these inputs! Surely this is flat violation of the product-technology assumption. But is it really? "Edible fats and oils" buys lots of the products primary to Vegetable oil mills. Thus, it is entirely possible to have two bottles of chemically identical oil made of identical raw materials by identical refining processes but with one bottle made entirely in Vegetable oil mills while the oil in the other bottle was pressed in those mills and then sold to Edible fats and oils for finishing. We might call this situation "trans-market product technology." Our algorithm gave the right answer for the Vegetable oil mills column of Recipe, that is, that the combined output of products primary to the oil mills with the inputs of oil sources which this industry had.

The fourth largest discrepancy in Table 10 is for the gasoline input into Automobile renting and leasing. Use shows $1131; Recipe ups that to $1197.2; but NewUse cuts it back to $565.5. What happened? The problem is that slightly more than half of the output Auto renting is produced in Credit agencies, with a minuscule input of gasoline. When NewUse is made, more than half of the gasoline in Recipe is allocated over to Credit agencies. Here we are confronted with a failure of the product-technology assumption not because of different processes for producing the same product but because two quite different products have been called one and the same in the accounting system. The output of the Credit agencies, long-term leasing, is quite distinct from the short-term renting, which is were the gasoline was used. The best solution would be to recognize the difference of the two products. Short of that, the worst of the problem can be fixed by turning the secondary transfer from Credit agencies to Automobile rental into a primary flow. The present Recipe matrix, incidentally, is about right in the gasoline row but fails to make a connection between a final demand for automobile renting and leasing and the output of credit agencies.

From these five or six cases, we see that our algorithm cannot be expected to give usable results on the first try. The problems, however, are likely to lie, however, neither in the fundamental economic reality nor in the algorithm, but in an accounting system which needs a few modifications in Use and Make to make it operational in our sense. Most importantly, the algorithm gives us the means to identify the places that need attention and a way of progressing systematically through the problems. It also provides a way of producing a final, non-negative Recipe matrix that implies a NewUse matrix close enough to the modified Use matrix that the differences can be safely ignored.

Making an input-output table requires fussing over details, and making a good Recipe matrix with the algorithm presented here is no different in this respect from any other part of the process. Use of the algorithm does, however, reveal where problems are. Moreover, the important problems are likely to be small in number. We have covered all of those causing a difference of as much as .100 between columns of Use and NewUse. To get to a Recipe table we would be ready to accept might another week's work. But in the total effort which went into making this table, that is minuscule. Most importantly, the use of the algorithm gives us a way to work on the problems rather than just wring our hands over negatives.

In this sense, this algorithm has performed satisfactorily over many years on every U.S. table since 1958. The use of the method seems to me to deserve to become a standard part of making input-output tables and, in particular, for making product-to-product tables based on the product-technology assumption.

5. The Computer Program

The C++ code for this algorithm, using functions from BUMP, the Beginner's Understandable Matrix Package, for handling matrices and vectors, is given below. The program and the supporting BUMP code made be downloaded from the Inforum Internet site: www.inforum.umd.edu. The main program here reads in the matrices that were used in the examples. The main program for the actual calculations of the full-scale American matrices is significantly larger and has various diagnostic output, such as that shown in Table 10.

In using the algorithm, it is important for documenting what has been done to have a method of input of the original Use and Matrix matrices that preserves the original version at the top of the input file and introduces the modifications as over-rides later in the file. It is also important to have software, such as MatView, which will show corresponding columns of several large matrices side-by-side in a scrolling grid. MatView is also available on the Inforum web site.

```
#include <stdio.h>        // for printf();
#include <math.h>         // for abs()
#include "bump.h"
int purify(Matrix& R, Matrix& U, Matrix& M, float toler);


void main(){
    Matrix Use(5,5), Make(5,5), R(5,5), NewUse(5,5);
    Use.ReadA("Use.dat");
    Make.ReadA("Make.dat");
    purify(R,Use,Make,.000001);
    R.Display("This is R");
    writemat(R,”Recipe”);
    NewUse = R*(~NewUse);
    writemat(NewUse,”NewUse”);
    tap();
    printf("\nEnd of calculations.\n");
    }

/* Purification produces a produt-to-product (or Recipe) matrix R from
    a Use matrix U and a Make matrix M.  M(i,j) shows the fraction of
    product j made in industry i.  U(i,j) shows the amount of product i
    used in industry j.  The product-technology assumption leads us to
    expect that there exits a matrix R such that U = RM'.  If, however, we
```

```
    compute R = U*Inv(M') we often find many small negative elements in
    R.  This routine avoids those small negatives in an iterative process.
    */

int purify(Matrix& R, Matrix& U, Matrix& M, float toler){
    int row, i, j, m, n, iter, imax;
    const maxiter = 20;
    float sum,rob,scale,dismax,dis;
    n = U.rows();  // n = number of rows in U
    m = U.columns(); // m = number of columns in U
    Vector C(m), P(m), Flow(m), Discrep(m);
    // Flow is row of U matrix and remains unchanged.
    // P becomes the row of the purified matrix.
    // C is the change vector at each iteration.
    // At the end of each iteration we set P = Flow + C, to start the next iteration.

    // Purify one row at a time
    for(row = 1; row <= n; row++){
        C.set(0.); // C, which will receive the changes, is initialized to zero.
        // P = Flow + C will be the new P.
        pulloutrow(Flow,U,row);
        P = Flow;
        iter = 0;
        start: iter++;
        for(j = 1; j<=m; j++){
            // Calculate total claims from other industries on
            // the inputs into industry j.
            sum = 0;
            for(i = 1; i <= m; i++){
                if(i == j) continue;
                rob = P[i]*M(j,i);
                sum += rob;
                C[i] += rob;
                }
            // Did we steal more from j than j had?
            if (sum > Flow[j] && sum > 0){
                // scale down robbery
                scale = 1. - Flow[j]/sum;
                for(i = 1; i <= m; i++){
                    if(i == j) continue;
                    C[i] -= scale*P[i]*M(j,i);
                    }
                sum = Flow[j];
                }
            C[j] -= sum;
            }
        // Check for convergence
        imax = 0;
        dismax = 0;
        for(i = 1; i <= m; i++){
            dis = fabs(P[i] - Flow[i] - C[i]);
            Discrep[i] = dis;
            if(dis >= dismax){
                imax = i;
                dismax = dis;
                }
            }
        P = Flow + C;
```

```
        C.set(0);
        if(dismax > toler){
            if(iter < maxiter) goto start;
            printf("Purify did not converge for row %d. Dismax = %7.2f. Imax = %d.\n",
                row,dismax,imax);
            }
        putinrow(P,R,row);
        }
    return(OK);
    }
```

References

Almon, C., Buckler, M., Horwitz, L., and Reimbold, T.,(1974) *1985, Interindustry Forecasts of the American Economy* (Lexington, Lexington Books) pp.151-154.

Rainer, N. and Richter, J. (1992) Some Aspects of the Analytical Use of Descriptive Make and Absorption Tables. *Economic Systems Research*, 4, pp.159 - 172

Steenge, A.E. and Konijin. A new Approach to Irreducibility in Multisectoral Models with Joint Production. *Economic Systems Research*, 4, pp 125-132

The *System of National Accounts 1993* (published by the United Nations, the World Bank, the IMF, the OECD, and the European Union)

ten Raa, Thijs (1988) An Alternative Treatment of Secondary Products in Input-Output Analysis:Frustration", *Review of Economics and Statistics, pp. 88-97.*

Van Rijckeghem (1967) "An Exact Mehod for Determining the TechnologyMatrix in al Situation with Secondary Products," *Review of Economics and Statistics*, pp. 607-8.