Commodity-by-Commodity Input-Output Matrices: Extensions and Experiences from an Application to Austria

Wolfgang Koller*
Industriewissenschaftliches Institut
Email: koller@iwi.ac.at

September 11, 2006

1 Introduction

The construction of commodity-by-commodity input-output matrices from Make and Use matrices is a much debated area. Different views compete with each other, both in theoretical aspects as in practical appraisement of the importance of the various issues involved.

The commodity-technology assumption and the industry-technology assumptions form the basic theoretical concepts between which one must choose. In an axiomatic framework Kop Jansen and ten Raa (1990) have shown that the compilation method based on the commodity-based assumption is superior (see also ten Raa and Rueda-Cantuche, 2003, for a recent overview). However, other strong arguments are in favor of the industry technology assumption, in particular the

*This work is part of the project "Austria in the World Economy: The Austrian Member of the INFORUM World Model", financed by the Jubiläumsfonds of the Austrian National Bank, which is gratefully acknowledged. The author thanks Bernhard Böhm and Josef Richter for valuable comments and guiding help. This work benefited also from direct or indirect contributions from them and other team members of the project group and colleagues at IWI, in particular Mikulas Luptacik, Bernhard Mahlberg and Sandra Lengauer, which are also thanked.

problem of how to avoid negative elements in the matrices when using the commodity technology assumption. It is also possible to mix both assumptions, using hybrid assumptions. Armstrong (1975) is a classical reference for hybrid models, Bohlin and Widell (2006) is a very recent one.

Almon (2000) presents an algorithm for the construction of commodity-by-commodity input-output matrices that solves the problem of negative elements by allowing deviations from the small commodity technology assumption or, viewed from another perspective, by correcting the Use matrix. In his paper Almon includes a discussion on the economic interpretation of the algorithm. Within the INFORUM family there exists a large body of experience with the use of Almon's algorithm (see Parve, 2004, for a recent application).

The present paper has Almon (2000) as a starting point and aims to deal with problems that arise when a whole set of input-output-tables must be compiled in a consistent way. This is normally the case in larger input-output-based projects, in particular in multisectoral macro models such as INFORUM models.

A task that is very complicated is the construction of a consistent set of Total Flow, Import-Flow and Domestic Flow matrices.[1] The (elementwise) sum of Import-Flow and Domestic Flow matrix must be equal to the Total Flow matrix. This imposes a serious restriction. To present and analyse different ways to deal with this restriction forms the primary motivation of this paper. For the construction of the set of flow matrices one can proceed bottom-up, top-down or based on forming the differenced-based. Different concepts can be used as guiding principles for these procedures and will be described in this paper. To our knowledge there are no prior academic publications that deal in depth with this issue.

Other problems arise when the Value Added matrix and the Employment matrix must be constructed on a commodity basis, given data on activity-basis. Apart from the avoidance of negative elements, other consistency requirements within the matrices to be calculated must be fulfilled. For example the number of full-time equivalents must be smaller than the number of jobs. An important issue is also the IO balance equation, which requires that the column sums of the Total Flow matrix plus the column sums of the commodity-based Value Added matrix must sum to the vector of produced commodities.

The aim of the paper is to present methods and practical procedures for the construction of a consistent set of commodity-by-commodity input-output tables. An application to Austria demonstrates their practicability. The application is part of the new Austrian INFORUM model, which is introduced in a companion paper (Böhm and Richter, 2006).

In the following two sections we develop methods for the construction of commodity-by-commodity matrices starting from Make and Use tables. Section 2 presents

[1] The most common name for the Total Flow Matrix is Transactions matrix. Almon (2000) uses the term Recipe matrix. In this paper we use the terms Total Flow matrix, Import-Flow matrix and Domestic Flow matrix to stress the relatedness between them. They are subsumed under the term flow matrix.

a generalized and an enhanced version of Almon's algorithm. Section 3 introduces concepts and methods for constructing Import-Flow and Domestic Flow matrices. Section 4 contains an application to Austria. While the application puts an emphasis on the tasks where the new methods proposed in Section 2 and 3 play a major role, other important tasks involved in the construction of input-output tables are treated as well. Section 5 concludes.

2 Algorithms for the construction of commodity-by-commodity tables

2.1 A description of the algorithm by Almon (2000)

Almon (2000) presents a way of making commodity-by-commodity tables from Use and Make matrices based on the commodity-technology assumption. He shows that slight adjustments in the commodity technology assumption can avoid negative elements in the Flow matrix. Before proposing useful extensions of Almon's algorithm we give a short description of it, using basically the notation of the original paper but relying more on matrix based notation.

Let us introduce some matrices and vectors. We assume an economy with n commodities produced by n activities. $\mathbf{U} = (u_{ij})$ is the Use matrix. Its elements specify the quantity of commodity i which is used as input by activity j. $\mathbf{V} = (v_{jk})$ is the Make matrix. Its elements denote the quantity of commodity k that is produced by activity j. From these two matrices one can derive the vector $\mathbf{x}' = (x_1, x_2, \ldots, x_n), x_k = \sum_{j=1}^{n} v_{jk}$ of produced commodities and the matrix $\mathbf{M} = (m_{jk}) = (v_{jk}/x_k)$, whose elements give the share of activity j in the production of commodity k. The aim is constructing the Flow matrix $\mathbf{R} = (r_{ik})$ that specifes the quantity of commodity i which is used in the economy in order to produce the commodity k. Finally, the matrix of technical input coefficients is given by $\mathbf{A} = (a_{ik}) = (r_{ik}/x_k)$.

With the commodity-technology assumption we have

$$\mathbf{R} = \mathbf{U}(\mathbf{M}')^{-1} \tag{1}$$

The construction of $\mathbf{R}$ according to (1) can also be performed row-wise:

$$\mathbf{r} = \mathbf{u}(\mathbf{M}')^{-1} \tag{2}$$

where $\mathbf{r}$ and $\mathbf{u}$ are rows of $\mathbf{R}$ and $\mathbf{U}$ with corresponding row index. ($\mathbf{r}$ and $\mathbf{u}$ are defined as row-vectors.) A simple iterative procedure

$$\mathbf{r}^{(l+1)} = \mathbf{r}^{(l)}(\mathbf{I} - \mathbf{M}') + \mathbf{u}, \tag{3}$$

initialized with $\mathbf{r}^{(0)} = \mathbf{u}$ will guarantee that $\mathbf{r}^{(l)}$ converges to $\mathbf{r}$ as long as the diagonal elements of $\mathbf{M}$ dominate the off-diagonal elements in a certain way.[2] Equation (3) serves as a starting point for Almon's algorithm.

We can rewrite (3) as

$$\mathbf{r}^{(l+1)} = \mathbf{u} - \mathbf{r}^{(l)}\check{\mathbf{M}}' + \mathbf{r}^{(l)}(\mathbf{I} - \hat{\mathbf{M}}'), \tag{4}$$

where $\hat{}$ denotes diagonalization by suppression of the off-diagonal elements of a square matrix and $\check{}$ denotes off-diagonalization by suppression of the diagonal entries of a square matrix. (Thus, $\mathbf{M} = \hat{\mathbf{M}} + \check{\mathbf{M}}$.)

It can be shown that in equation (4) the third term on the right-hand side can be replaced by $(\mathbf{e}'\check{\mathbf{M}}) \otimes \mathbf{r}^{(l)}$, where $\mathbf{e}$ is a summation vector and $\otimes$ denotes elementwise multiplication of two matrices or vectors of the same dimension (elementwise multiplication is also known as Hadamard or Schur product):

$$\mathbf{r}^{(l+1)} = \mathbf{u} - \mathbf{r}^{(l)}\check{\mathbf{M}}' + (\mathbf{e}'\check{\mathbf{M}}) \otimes \mathbf{r}^{(l)}, \tag{5}$$

We can see that in (5) the new $\mathbf{r}^{(l+1)}$ is formed by starting with $\mathbf{u}$ and then subtracting something (the second term on the right-hand side) and adding something (the third term on the right hand side). What is added and was is substracted depends on the current technological assumptions on the use of the commodity considered, represented by $\mathbf{r}^{(l)}$. Almon (2000) gives an economic interpretation of this iteration process.

Almon's algorithm introduces "stops" in equation (5) that prevent elements from becoming negative. Whenever more is about to be subtracted from an element of $\mathbf{u}$ then the corresponding element of the intended substraction term is scaled down such that $\mathbf{r}^{(l+1)}$ is not negative. Then, in the addition term on the right-hand side of (5) this scaling down factors must also be considered in an appropriate way. With these modifications the iteration formula of Almon's algorithm becomes

$$\mathbf{r}^{(l+1)} = \mathbf{u} - \mathbf{s}^{(l)} \otimes \mathbf{w}^{(l)} + (\mathbf{s}^{(l)}\check{\mathbf{M}}) \otimes \mathbf{r}^{(l)}, \tag{6}$$

where $\mathbf{s}^{(l)}$ is a row-vector of "stops" or scaling factors and $\mathbf{w}^{(l)} = \mathbf{r}^{(l)}\check{\mathbf{M}}'$.

Including the iteration formula for $\mathbf{s}^{(l)}$ and arranging all necessary steps, Almon's algorithm can be defined as follows:

(i) Set $i = 1$ (Start with first row),

(ii) Set $l = 0$ and $\mathbf{r}^{(l)} = \mathbf{u} = \mathbf{u} =$ i-th row of $\mathbf{U}$ (Start iterative procedure),

(iii) Set $\mathbf{w}^{(l)} = \mathbf{r}^{(l)}\check{\mathbf{M}}'$,

(iv) Set row-vector $\mathbf{s}^{(l)}$ such that each of its elements $s_k^{(l)}$ satisfies:

$$s_k^{(l)} = \begin{cases} 1 & \text{if } u_k < w_k^{(l)} \\ u_k/w_k^{(l)} & \text{otherwise} \end{cases}$$

[2] see Almon (2000) for more details.

(v) Set $\mathbf{r}^{(l+1)} = \mathbf{u} - \mathbf{s}^{(l)} \otimes \mathbf{w}^{(l)} + (\mathbf{s}^{(l)}\check{\mathbf{M}}) \otimes \mathbf{r}^{(l)}$,

(vi) Test for convergence by comparing $\mathbf{r}^{(l+1)}$ and $\mathbf{r}^{(l)}$,

(vii) If convergence has occurred assign $\mathbf{r}^{(l)}$ to the i-th row of $\mathbf{R}$, otherwise set $l = l + 1$ and perform steps (iii)–(vi) again,

(viii) If $i = n$ stop, otherwise set $i = i + 1$ and perform steps (ii)–(vii) again.

This definition of Almon's algorithm uses row-wise notation, which seems natural in view of its implementation as a computer program. The iterations necessary until convergence will probably differ for each row and efficient computer programming should consider this. However, the definition of the algorithm in matrix notation is also revealing:

$$\mathbf{R}^{(l+1)} = \mathbf{U} - \mathbf{S}^{(l)} \otimes (\mathbf{R}^{(l)}\check{\mathbf{M}}') + (\mathbf{S}^{(l)}\check{\mathbf{M}}) \otimes \mathbf{R}^{(l)}, \tag{7}$$

where we skip the iteration formula for $\mathbf{S}^{(l)}$. After convergence we have

$$\mathbf{R}^{*} = \mathbf{U} - \mathbf{S} \otimes (\mathbf{R}^{*}\check{\mathbf{M}}') + (\mathbf{S}\check{\mathbf{M}}) \otimes \mathbf{R}^{*}. \tag{8}$$

The economic interpretation of the result of the algorithm can be viewed from two perspectives. Differences between $\mathbf{R}^{*}$ and $\mathbf{R}$ can be ascribed either to deviations from the commodity technology assumption or to "errors" in the Use matrix. A corrected "New Use" matrix is implied by the Total Flow matrix delivered by the algorithm as

$$\mathbf{U}^{*} = \mathbf{R}^{*}\mathbf{M}'. \tag{9}$$

The inspection of this "New Use" matrix possibly yields valuable information on problem areas in the original data. In section 3 we will need this matrix as an ingredient for construction formulas for consistent Import-Flow and Domestic Flow matrices.

2.2 Generalisation of Almon's algorithm

To demand from the Flow Matrix that it contains only non-negative elements is descretionary in a certain sense. Any lower bound other than zero could be justified as well. In the course of the construction of a consistent set of input-output tables we found out that there is a need for a modification of Almon's algorithm to deal with general lower-bound matrices. For example a-priori information from older input-output tables or consistency requirements may provide minimum values for specific elements of the commodity-by-commodity matrix.

We developed such a modification of Almon's algorithm that integrates the information on lower-bound restrictions defined by the lower-bound matrix $\mathbf{B} = (b_{ik})$. The modified algorithm is a generalisation of Almon's algorith because it implements the conventional Almon's algorithm when a lower-bound matrix $\mathbf{B} = \mathbf{0}$ is supplied, wher $\mathbf{0}$ is the zero-matrix.

The Generalized Almon's algorithm can be defined as follows:

(i) Set $i = 1$ (Start with first row),

(ii) Set $l = 0$, $\mathbf{r}^{(l)} = \mathbf{u} = i$-th row of $\mathbf{U}$ and $\mathbf{b} = i$-th row of $\mathbf{B}$ (Start iterative procedure),

(iii) Set $\mathbf{w}^{(l)} = \mathbf{r}^{(l)}\check{\mathbf{M}}'$,

(iv) Set row-vector $\mathbf{s}^{(l)}$ such that each of its elements $s_k^{(l)}$ satisfies:

$$s_k^{(l)} = \begin{cases} 1 & \text{if } u_k - w_k^{(l)} < b_k \\ (u_k - b_k)/w_k^{(l)} & \text{if } 1 > (u_k - b_k)/w_k^{(l)} > 0 \\ 0 & \text{otherwise} \end{cases}$$

(v) Set $\mathbf{r}^{(l+1)} = \mathbf{u} - \mathbf{s}^{(l)} \otimes \mathbf{w}^{(l)} + (\mathbf{s}^{(l)}\check{\mathbf{M}}) \otimes \mathbf{r}^{(l)}$,

(vi) Test for convergence by comparing $\mathbf{r}^{(l+1)}$ and $\mathbf{r}^{(l)}$,

(vii) If convergence has occurred assign $\mathbf{r}^{(l)}$ to the i-th row of $\mathbf{R}$, otherwise set $l = l + 1$ and perform steps (iii)–(vi) again,

(viii) If $i = n$ stop, otherwise set $i = i + 1$ and perform steps (ii)–(vii) again.

Just as the conventional Almon's algorithm this algorithm converges as long as equation (3) converges. However, in order to receive meaningful results it is necessary to supply a meaningful lower-bound matrix to the algorithm. An obvious example for a not very meaningful lower-bound matrix would be $\mathbf{B} = \mathbf{U}$, which would result in $\mathbf{R}^* = \mathbf{U}$ in one single iteration.

It is also possible that the lower-bound conditions specified by $\mathbf{B}$ cannot be satisfied, if some or all elements are chosen too large. Convergence is nevertheless guaranteed because we restricted $s_k^{(l)}$ to be non-negative in step (iv) of the iteration. This restriction seems economically reasonable since the direction of the reallocations of the flows in the iteration procedure in step (v) of the algorithm must not be reversed, only restricted.

2.3 Enhancement of Almon's algorithm

With the Almon's algorithm it can happen that a specific element of $\mathbf{R}^*$ is positive even though the corresponding element of $\mathbf{R}$ is negative. Or, in the Generalised Almon's algorithm it can happen that $r_{ik}^* > b_{ik}$ even though $r_{ik} < b_{ik}$. It seems desirable to ensure that $r_{ik}^* = b_{ik}$ if $r_{ik} < b_{ik}$. The lower bound conditions should be exactly met in those cases where they are needed. This suggests another modification of the algorithm.

The Enhanced Almon's algorithm introduces an emdedded iteration procedure within the iteration formula for $s_k^{(l)}$ in step (iv) of the algorithm. The exact specification of the Enhanced Almon's algorithm is available from the author upon request.

3 Methods for the construction of import-flow matrices

3.1 Notation and Definitions

Before we can present and discuss several ways of constructing a consistent set of symmetric matrices for total flows, domestic flows and import flows, we need to introduce some notation, in addition to the one use in the previous section. Furthermore we present a definition of the well-known commodity technology assumption and of two alternative assumptions concerning import-proportions that will form the theoretical guides for the methods developed.

Let $\mathbf{Z} = (z_{ijk})$ be a three-dimensional array whose elements define the quantity of commodity i which is used as input by activity j for the production of commodity k. We could call $\mathbf{Z}$ the Flow-Use-System array. If $\mathbf{Z}$ were known (of course, in practical applications it is not), then the Use and Flow matrices could be easily derived as $\mathbf{U} = (u_{ij}) = \sum_{k=1}^{n} z_{ijk}$ and $\mathbf{R} = (r_{ik}) = \sum_{j=1}^{n} z_{ijk}$, respectively. Analogously, we define $\mathbf{Z}_m = (z_{ijk}^m)$ as the three-dimensional array of the quantity of imports of commodity i which is used as input by activity j for the production of commodity k (Import-Flow-Use-System array). Thus, the Import-Use matrix and Import-Flow matrix we denote as $\mathbf{U}_m = (u_{ij}^m) = \sum_{k=1}^{n} z_{ijk}^m$ and $\mathbf{R}_m = (r_{ik}^m) = \sum_{j=1}^{n} z_{ijk}^m$, respectively. The matrix of import-input coefficients is then given by $\mathbf{A}_m = (a_{ik}^m) = (r_{ik}^m / x_k)$. Similarly, we could define domestic versions of the above matrices, but need not do so, because they can be formed as difference (e.g., $\mathbf{A}_d = \mathbf{A} - \mathbf{A}_m$) and everything that is said in the following about the import-versions applies in a parallel way to the domestic versions.

We can formulate two kinds of import-proportion matrices for commodities used as inputs, one by activities and one by the produced commodities. The first, $\mathbf{P_U} = (p_{ij}^{\mathbf{U}}) = (u_{ij}^m / u_{ij})$, defines the share of imports of commodity i used as input by activity j in total quantity of commodity i used as input by activity j. The second, $\mathbf{P_R} = (p_{ik}^{\mathbf{R}}) = (r_{ik}^m / r_{ik})$, defines the share of imports of commodity i used as input for the production of commodity k in total quantity of commodity i used as input for the production of commodity k. (It should be noted that for these definitions it is necessary to define division of zero by zero as zero.)

Based on this notation we can define three assumptions:

Commodity Technology Assumption (CTA): Producing one unit of commodity k always requires the same quantity of commodity i as input, irrespective of the activity in which production is taking place:

$$z_{ijk}/v_{jk} = a_{ik} \text{ for all } i,\ j \text{ and } k$$

Commodity-specific Import-Proportionality Assumption (CSIPA): The share of imported inputs of commodity i in total inputs of commodity i which is used for production of commodity k is always the same, irrespective of the activity in which production is taking place:

$$z^m_{ijk}/z_{ijk} = p^{\mathbf{R}}_{ik} \text{ for all } i,\ j \text{ and } k$$

Industry-specific Import-Proportionality Assumption (ISIPA): The share of imported inputs of commodity i used by activity j in total inputs of commodity i used by activity j is always the same, irrespective of the commodity which is produced:

$$z^m_{ijk}/z_{ijk} = p^{\mathbf{U}}_{ij} \text{ for all } i,\ j \text{ and } k$$

It is not completely obvious which of CSIPA or ISIPA is to prefer from an economic point of view. The CSIPA has the merit of being conceptually near to the CTA: if one puts trust in the constancy of a_{ik} over different activities, why shouldn't one also trust in the constancy of the *composition* of a_{ik} over different activities? But there is certainly no technical necessity and every activity is free to substitute imported inputs for domestic inputs and the other way round. Examples are conceivable that speak for the ISIPA. If both agriculture and construction industry produce construction services and use sand as input, it is likely that the former will import a smaller proportion of it than the latter because the latter is probably nearer to import markets and transport ways. For many inputs the ISIPA will be a good description of reality. For example, when an activity produces different commodities and uses bureau machines as input in both production processes, it might well need different quantities of them for the production of one unit of the two produced commodities, but the share of imported bureau machines will probably be the same in both production processes. We think that the decision between CSIPA and ASIPA is mainly an empirical problem.

3.2 Construction of the import-flow matrix based on the CSIPA

The simultaneous validity of CTA and CSIPA amounts to applying the CTA to import flows, i.e. one assumes that producing one unit of commodity k always requires the same quantity of imports of commodity i as input, irrespective of the activity in which production is taking place: $z^m_{ijk}/v_{jk} = a^m_{ik}$ for all i, j and k. This is proven by transforming

$$\begin{aligned} z^m_{ijk}/v_{jk} &= (p^{\mathbf{R}}_{ij} z_{ijk})/v_{jk} = p^{\mathbf{R}}_{ij}(v_{jk}a_{ik})/v_{jk} = p^{\mathbf{R}}_{ij}a_{ik} = p^{\mathbf{R}}_{ij}(r_{ik}/x_k) \\ &= (r^m_{ik}/r_{ik})(r_{ik}/x_k) = r^m_{ik}/x_k = a^m_{ik}, \end{aligned} \tag{10}$$

where use is made of the CSIPA, the CTA and the defnitions of a_{ik}, $p^{\mathbf{R}}_{ij}$ and a^m_{ik} (in that order).

Thus, the simultaneous validity of CTA and CSIPA provides justification for using

$$\mathbf{R}_m = \mathbf{U}_m(\mathbf{M}')^{-1} \tag{11}$$

for the construction of the import-flow matrix.

However, when we meet a problem with negatives in the flow matrix, it is even likelier that the negatives appear in the import-flow matrix, too. Applying Almon's algorithm to construct the import-flow matrix solves the problem. We denote the import-flow matrix constructed with Almon's algorithm by $\mathbf{R}^*_m$. A small flaw of this approach is that it cannot be said whether the differences between $\mathbf{R}^*_m$ and $\mathbf{R}_m$ are caused by deviations from the CTA or by deviations from the CSIPA. More serious is the problem of inconsistency between the import-flow matrix, domestic-flow matrix and total flow matrix in that $\mathbf{R}^* \neq \mathbf{R}^*_m + \mathbf{R}^*_d$.

Therefore we propose an alternative method for the construction of the import-flow matrix that proceeds top-down. It takes the total flow matrix and then uses the CSIPA.

First let us derive the procedure for the case where there is no negatives problem by transforming (11) as follows:

$$\mathbf{R}_m = (\mathbf{P_U} \otimes \mathbf{U})(\mathbf{M}')^{-1} = (\mathbf{P_U} \otimes (\mathbf{RM}'))(\mathbf{M}')^{-1}, \tag{12}$$

where we have made use of the definition of $\mathbf{P_U}$ and of the CTA. For overcoming the negatives-problem we insert $\mathbf{R}^*$ instead of $\mathbf{R}$ and get

$$\mathbf{R}^*_m = (\mathbf{P_U} \otimes (\mathbf{R}^*\mathbf{M}'))(\mathbf{M}')^{-1} = (\mathbf{P_U} \otimes \mathbf{U}^*)(\mathbf{M}')^{-1}, \tag{13}$$

Note that equation (13) uses the "New Use" matrix, $\mathbf{U}^*$. The term $\mathbf{P_U} \otimes \mathbf{U}^*$, consequently could be called "New Import Use" and denoted by $\mathbf{U}^*_m$.[3]Equation (13) then would read as

$$\mathbf{R}^*_m = \mathbf{U}^*_m(\mathbf{M}')^{-1}, \tag{14}$$

The Import Flow matrix $\mathbf{R}^*_m$ constructed with equation (13 may still contain negative elements. Therefore it might be necessary to apply Almon's algorithm

[3]We also experimented with an alternative derivation of $\mathbf{U}^*_m$, which is based on solving equation (8) for $\mathbf{R}^*$, then substituting for $\mathbf{R}^*$ in (9) and finally substituting $\mathbf{U}_m$ for $\mathbf{U}$. Although this approach seems promising its development has not yet been completed.

on the level of the import flow matrix again — this time to correct for deviations from the CSIPA.

The proposed procedure does not meet the balance property since the row sums of $\mathbf{R}_m^*$ are not necessarily the same as those of $\mathbf{R}_m$. Rescaling each row will solve this problem.

3.3 Construction of the import-flow matrix based on the ISIPA

As we have motivated before, the CTA can be combined with the ISIPA without hurting any theoretical principles. Derivation of the import-flow matrix under the simultaneous validity of CTA and ISIPA is straightforward:

$$\begin{aligned} r_{ik}^m &= \sum_{j=1}^n z_{ijk}^m = \sum_{j=1}^n z_{ijk} p_{ij}^{\mathbf{U}} = \sum_{j=1}^n p_{ij}^{\mathbf{U}} v_{jk} a_{ik} = a_{ik} \sum_{j=1}^n p_{ij}^{\mathbf{U}} v_{jk} \\ &= (a_{ik} x_k) \sum_{j=1}^n p_{ij}^{\mathbf{U}} v_{jk} / x_k = r_{ik} \sum_{j=1}^n p_{ij}^{\mathbf{U}} m_{jk}, \end{aligned}$$

which can be expressed in matrix notation by

$$\mathbf{R}_m = \mathbf{A} \otimes (\mathbf{P_U V}) = \mathbf{R} \otimes (\mathbf{P_U M}). \tag{15}$$

When there is a problem with negatives and the total-flow matrix has been constructed with the help of Almon's algorithm, then simply insert $\mathbf{R}^*$ instead of $\mathbf{R}$ and get:

$$\mathbf{R}_m^* = \mathbf{R}^* \otimes (\mathbf{P_U M}). \tag{16}$$

Again, the proposed procedure has the property that the row sums of $\mathbf{R}_m^*$ are not the same as those of $\mathbf{U}_m$ and again a remedy is rescaling each row accordingly.[4]

4 Application: Construction of a consistent set of input-output tables for Austria

4.1 Description of the data and its preparation

The application is based on the Make-Use table for Austria 2001, as published by Statistik Austria in spring 2005.[5] This data source includes, among other information, sub-tables for imports and tables on value added (in the dimension activities

[4]An interesting detail is that the scaling factors for the the CSIPA and ISIPA must be the same. This is easy to see when considering row-wise formulations of equation (13) and (16). For the CSIPA the sum of the row vector $\mathbf{r}_m^*$ is $\mathbf{r}_m^*\mathbf{e} = (\mathbf{p_U} \otimes \mathbf{u}^*)(\mathbf{M}')^{-1}\mathbf{e} = (\mathbf{p_U} \otimes \mathbf{u}^*)\mathbf{e} = \mathbf{p_U}\mathbf{u}^{*\prime}$, where $\mathbf{e}$ is a summation vector. For the ISIPA we have $\mathbf{r}_m^*\mathbf{e} = \mathbf{r}^*(\mathbf{p_U M})' = \mathbf{r}^*\mathbf{M}'\mathbf{p}_\mathbf{U}' = \mathbf{u}^*\mathbf{p}_\mathbf{U}'$.

[5]Statistik Austria (2005), see also the webpage of Statistik Austria, http://www.statistik.at

by components) and employment (activities by components). The Austrian Make-Use system is a particularly challenging application field for the methods we want to compare because Statistik Austria is very consequent in the compilation of the tables. Therefore the Austrian Make matrix contains relatively many off-diagonal elements.

As a first step of the data preparation we carried out aggregation and disaggregation of some sectors.[6] This step was motivated by the peculiarities of the Austrian economy and by the intended capabilities of the Austrian INFORUM model (Böhm and Richter, 2006). The resulting classification has 56 sectors.[7]

After a first phase of experimentation with the data it was decided that there are two entries in the Make table that inhibit successful compilation of input-output, value added and employment tables on a commodity-by-commodity, components-by-commodity and categories-by-commodity basis, respectively. These "problematic areas" are the production of "Food products and beverages" by the activity "Products of agriculture and fishing", most of which is wine, and of "Chemicals, chemical products" by the activity "Coke, refined petroleum products". For example, the problems we encountered in these sectors after construction of the commodity-based tables included relatively large negative entries in the tables and average wages that differed strongly from the average wages on an activity basis. We isolated these areas from all the tables and separately translated them into commodity-based tables, i. e. we applied the ITA to these two entries of the Make table. At a later stage the tables were added to the other tables to form the commodity-based overall tables. The applications we discuss in the next two subsections concern the Make-Use tables we received after isolating the two problematic areas.

4.2 An overview of the problem of negatives

It is mainly the problem with negative flows that causes all the troubles this part of our application has to deal with. Therefore in this subsection we want to give an overview of the relevance of that problem. In Table 1 we report information on the number and other summary statistics of the negative elements in the Total Flow matrix $\mathbf{R}$, the Import-Flow matrix $\mathbf{R}_m$ and the Domestic Flow matrix $\mathbf{R}_d$ constructed with equations 1 and 11 and $\mathbf{R}_d = \mathbf{R} - \mathbf{R}_m$.

[6]CPA 01 "Products of agriculture, hunting" and CPA 05 "Fish, other fishing products" were aggregated to form a new sector "Products of agriculture and fishing". CPA 60 "Land transport and transport via pipeline services" and CPA 61 "Water transport services" were aggregated into "Land and water transport and transport via pipeline services". Using more detailed tables published by Statistik Austria CPA 40 was disaggregated into CPA 40.1 "Electrical energy" and the residual sector "Gas, steam and hot water".

[7]See the Appendix for a listing of the sector names and used abbreviations. For simplicity we use only the names from the commodity classification. It will be clear from the context when the corresponding activity is meant.

Table 1: Negative elements in Total, Import and Domestic Flow matrices

	Total flow matrix		Import flow matrix		Domestic flow matrix	
N.(< 0)		433		620		448
N.(< −1000)		98		105		55
Sum		563252 (0.32)		527857 (0.95)		280957 (0.23)
Row sums:						
Top 1	Wood	80106 (2.18)	BasMet	71932 (1.67)	Wood	51032 (1.79)
Top 2	SeCuSp	49627 (2.49)	SeCuSp	71259 (8.56)	GasSHD	29351 (1.79)
Top 3	BasMet	45284 (0.72)	Constr	35387 (6.43)	AgricF	28730 (0.75)
Top 4	CruOre	40778 (1.25)	MachEq	34765 (0.86)	BasMet	23958 (1.23)
Column sums:						
Top 1	MetPrd	73206 (1.55)	Wholes	78795 (5.08)	ElecD	37151 (1.44)
Top 2	RadCEq	69077 (2.24)	RadCEq	51032 (2.29)	RadCEq	35872 (4.15)
Top 3	ElecD	47396 (1.27)	MetPrd	44847 (2.43)	MetPrd	34137 (1.19)
Top 4	Retail	40022 (0.70)	Retail	43139 (6.86)	Wholes	25916 (0.31)

All sums are in 1000 EUR. Percentages are in parentheses. Sums and percentage values should be read as negative values. See the Appendix for the long names of the commodities.

The analysis summarized in Table 1 shows that negative entries pose a problem of varying severity depending on the flow matrix considered and whether one looks at selected rows or columns.[8] In the Import-Flow matrix, the problem of negative elements turns out to be the severest, with 620 negative elements of 3136. In the Total Flow and Domestic Flow matrix 433 and 448 elements, respectively, are negative. The Import-Flow matrix surpasses the other two also when only elements smaller than -1000 EUR are counted. The sum of the negative elements of the Import-Flow matrix, -528 Mill. EUR comes near the corresponding figure for the Total Flow matrix. While -0.95 percent of the sum of the Import-Flow matrix are found in negative elements it is -0.23 percent for the Total Flow matrix and -0.32 for the Domestic Flow matrix.

This suggests that deviations from CSIPA account for *at least* two thirds of the sum of negative elements in the Import Flow matrix and deviations from the CTA account for *at most* one third. This is because in the Import-Flow matrix some of the deviations from the CSIPA could cancel out the effect of deviations from the CTA. In the case of the Domestic Flow matrix, some of the effects of deviations from the CTA must have been cancelled out by deviations from the CSIPA, otherwise the percentage of the sum of negative elements could not be lower (absolutely) than for the Total Flow matrix.

[8] Of course, negative elements in the flow matrix can also indicate a problem on the level of individual cells but this aspect is left aside.

4.3 Construction of consistent Total Flow, Import-Flow and Domestic Flow matrices

In this subsection we describe four alternative approaches we used for the construction of a set of consistent Total Flow, Import-Flow and Domestic Flow matrices. In all of them we employed the Enhanced Almon's algorithm in order to avoid negatives and to incorporate a-priori information in the form of lower bounds for the elements of the flow matrix.

Besides a-priori information gained from previous steps in the procedure, this included also a-priori information from common knowledge of production processes and from published input-output-tables for preceding years. A detailed examination of the cells of the flow matrices returned by Almon's algorithm revealed that many elements were zero, although a-priori information suggested a "small" positive value. Based on a moderate industry technology assumption we calculated minimum values for these cells and looked for a way to ensure lower bounds for the commodity-by-commodity matrix. The Enhanced Almon's algorithm using a lower bounds matrix $\mathbf{B}$ was found to handle this problem well.[9] The lower bounds matrices for the Total, Import and Domestic Flow matrices contain 12, 4 and 9 elements, respectively, that were greater 1000 EUR. The sums are 53, 13, 40 Mill EUR, respectively. The maximum values of the lower bound matrices are 7.7, 2.1 and 7.7 Mill. EUR, respectively.

For the construction of a set of Total Flow, Import-Flow and Domestic Flow matrices[10] satisfying $\mathbf{R} = \mathbf{R}_d + \mathbf{R}_m$ we worked out four different approaches:

- bottom-up approach (approach A)
- difference-based approach (approach B)
- top-down approach based on the CSIPA (approach C)
- top-down approach based on the ISIPA (approach D)

The bottom-up approach (A) takes $\mathbf{U}_m$ and $\mathbf{U}_d$ and uses the Enhanced Almon's algorithm and lower bound matrices $\mathbf{B}_m$ and $\mathbf{B}_d$ to separately calculate $\mathbf{R}_m$ and $\mathbf{R}_d$. Then the Total Flow matrix is calculated as $\mathbf{R} = \mathbf{R}_m + \mathbf{R}_d$. It is guaranteed to have no negative elements. In this approach we use the Enhanced

[9] As an alternative and for quality control purposes we also proceeded as follows: supply $\mathbf{U} - \mathbf{B}$ to Almon's algorithm and then add $\mathbf{B}$ again. This procedure worked about equally well in comparison with the enhanced Almon's algorithm. In the case of the Total Flow matrix the lower bound criteria were fulfilled in an equal manner, concerning those cells where the lower bound restrictions were effective. In other cells, a total of 3.1 Mill EUR (without double counting) were allocated in different cells than in the Total Flow matrix constructed with the Enhanced Almon's algorithm.

[10] While in sections 2 and 3 we used the asterisk to indicate that a Flow matrix was constructed with the help of Almon's algorithm, here we drop this notation. It will be clear from the context which kind of construction method the respective Flow matrix is based on.

Almon's algorithm to take account not only of deviations from the CTA but also of deviations from the CSIPA. This might be seen as a disadvantage. However, using the CTA always implies the assumption of homogeneity of product groups, which sometimes is far from reality. Imported and domestic goods can be totally different even though classified in the same product group. Cotton provides a convincing example. Austria is an importer of cotton and unable to substitute own production for these imports. Therefore it is practical to interpret imported and domestic goods as different goods in the construction of $\mathbf{R}$, $\mathbf{R}_m$ and $\mathbf{R}_d$. Another advantage of approach A is its great simplicity.

The difference-based approach (B) is somewhat more complicated. It starts with either $\mathbf{U}_m$ or $\mathbf{U}_d$. Let us assume we start with $\mathbf{U}_m$ and denote this variant with superscript (m). Approach B then uses the enhanced Almon's algorithm and $\mathbf{B}_m$ to calculate $\mathbf{R}_m^{(m)}$. Then a new lower bound matrix $\mathbf{B}^{(m)}$ is calculated whose elements are the maximum of the corresponding elements in $\mathbf{B}_m$ and $\mathbf{R}_m$. A first variant of the Total Flow matrix $\mathbf{R}^{(m)}$ is calculated based on the Enhanced Almon's algorithm and $\mathbf{B}^{(m)}$. We get the first variant of the Domestic Flow matrix as difference, $\mathbf{R}_d^{(m)} = \mathbf{R}^{(m)} - \mathbf{R}_m^{(m)}$. Second variants of the Total, Import and Domestic Flow matrices, $\mathbf{R}^{(d)}$, $\mathbf{R}_m^{(d)}$ and $\mathbf{R}_d^{(d)}$, are gained by starting with $\mathbf{U}_d$ and proceding analogously as before. We arrive at the final versions by forming averages $\mathbf{R} = 0.5(\mathbf{R}^{(m)} + \mathbf{R}^{(d)})$, $\mathbf{R}_m = 0.5(\mathbf{R}_m^{(m)} + \mathbf{R}_m^{(d)})$ and $\mathbf{R}_d = 0.5(\mathbf{R}_d^{(m)} + \mathbf{R}_d^{(d)})$.

In the practical application of approach B a further complication arose because in the calculation of $\mathbf{R}^{(d)}$ the Enhanced Almon's algorithm was not able to fulfill the lower bound restrictions. In 10 cases the algorithm missed it by more than 1000 EUR and in 5 cases by more than 100000 EUR. With a sum of 13.6 mill. EUR (0.007 percent of the sum of the Total Flow matrix) we rated this problem as unimportant and solved it by proportional row-wise redistribution.

The top-down approach based on the CSIPA (approach C) starts with the calculation of $\mathbf{R}$, using the Enhanced Almon's algorithm and lower-bound matrix $\mathbf{B}$. Then we can proceed either with the calculation of a first variant of the import flow matrix or a first variant of the Domestic Flow matrix. Let us assume we proceed with the former and denote this variant with superscript (m). We calculate a new Import-Use matrix $\mathbf{U}_m^{(m)} = \mathbf{P}_\mathbf{U} \otimes (\mathbf{R}\mathbf{M}')$ and then rescale each row of $\mathbf{U}_m^{(m)}$ such that its row sums are equal to the row sums of $\mathbf{U}_m$.[11] Then $\mathbf{R}_m^{(m)}$ is calculated with the Enhanced Almon's algorithm, using $\mathbf{U}_m^{(m)}$ instead of $\mathbf{U}_m$ and using $\mathbf{B}_m$. [12] The first version of the Domestic Flow matrix is obtained as difference, $\mathbf{R}_d^{(m)} = \mathbf{R} - \mathbf{R}_m^{(m)}$. Proceeding analogously we calculate second versions

[11]As mentioned in section 3 the scaling factors must be the same for approach C and D, see the discussion of approach D for more information on the scaling factors.

[12]It was necessary to use the Enhanced Almon's algorithm a second time because negative elements appeared when using $\mathbf{R}_m^{(m)} = \mathbf{U}_m^{(m)}(\mathbf{M}')^{-1}$. However, the sum of negative elements in $\mathbf{R}_m^{(m)}$ was only 0.73 percent of the sum of all elements of $\mathbf{R}_m^{(m)}$, i.e. 0.22 percent points less than based on the original Import-Use matrix.

of Domestic and Import-Flow matrices, $\mathbf{R}_d^{(d)}$ and $\mathbf{R}_m^{(d)}$. As before with approach B, we arrive at the final versions by forming averages of the (m)- and (d)-versions of the matrices.

In the practical application of approach C we met some obstacles. Some elements of $\mathbf{R}_m^{(m)}$ were larger than the corresponding elements of $\mathbf{R}$, which would have implied negative elements in $\mathbf{R}_d^{(m)}$. The same problem also showed up in $\mathbf{R}_d^{(d)}$. In the case of $\mathbf{R}_m^{(m)}$ this problem made up 0.04 percent of the sum of all elements. In the case of $\mathbf{R}_d^{(d)}$ it was 0.26 percent. We solved the problem by proportional row-wise redistribution of the offending amounts.[13]

Like approach C, the top-down approach based on the ISIPA (approach D) starts with the calculation of $\mathbf{R}$ and then proceeds with either the calculation of a first variant of the Import-Flow matrix or of the Domestic Flow matrix. Assuming we start with the former and using the superscript (m) for this variant, we calculate $\mathbf{R}_m^{(m)}$ using equation (16). We need to rescale each row of $\mathbf{R}_m^{(m)}$ such that its row sums are equal to the row sums of $\mathbf{U}_m$. The first version of the Domestic Flow matrix is obtained as difference, $\mathbf{R}_d^{(m)} = \mathbf{R} - \mathbf{R}_m^{(m)}$. Proceeding analogously we calculate second versions of Domestic and Import-Flow matrices, $\mathbf{R}_d^{(d)}$ and $\mathbf{R}_m^{(d)}$. As with approaches B and C, we arrive at the final versions by forming averages of the (m)- and (d)-versions of the matrices.

The practical application of this approach did not pose major problems. The factors for rescaling ranged from 0.9984 to 1.0388 for the (m)-version and from 0.9780 to 1.0304 for the (d)-version. Their means were 1.0034 and 1.0003, respectively. After rescaling some elements of the rescaled matrix were larger than the corresponding elements in $\mathbf{R}$. To resolve this, we redistributed the offending amounts on the other cells in the same rows. However, the sum of redistributed flows was almost negligible.

4.4 Comparison of Total Flow, Import-Flow and Domestic Flow matrices constructed with the four different approaches

In this subsection we compare the Total Flow, Import-Flow and Domestic Flow matrices constructed with the four different approaches. The aim of this comparison is merely to say how near to each other the different approaches are. Not knowing the "true" flow matrices there is no general way to say whether any approach is better than the other.

The comparison of the differences of the matrices rendered by different approaches can be carried out on the level of individual rows, columns or even individual cells. On this level it might be possible to identify inplausibilities in

[13] Motivated by this problem, we also experimented with a further modification of Almon's algorithm that integrated upper-bound restrictions. But this idea turned out not to be practicable.

Table 2: Comparison of the various approaches for construction of Total Flow, Import-Flow and Domestic-Flow matrices

	Total flow matrix		Import flow matrix	Domestic flow matrix	
Comparison	(1)	(2)	(1)	(1)	(2)
A vs. B	267922 (0.15)	0.0014	56344 (0.10)	223388 (0.18)	0.0007
A vs. C	461455 (0.26)	0.0025	165416 (0.30)	447022 (0.36)	0.0014
A vs. D	461455 (0.26)	0.0025	1493212 (2.70)	1824291 (1.48)	0.0043
B vs. C	216667 (0.12)	0.0011	125722 (0.23)	244253 (0.20)	0.0008
B vs. D	216667 (0.12)	0.0011	1499119 (2.71)	1643925 (1.34)	0.0038
C vs. D	0 (0.00)	0	1542748 (2.79)	1542748 (1.25)	0.0036

(1) Sum of absolute differences between the two matrices over all elements, sums are in 1000 EUR, percentages are in parentheses. (2) Mean absolute differences of final demand multipliers between the models implied by the two matrices.

the results for one or all compared approaches. Problem-areas in the data might be detected, that motivate returning to the data preparation phase, applying the industry technology assumption to selected cells, introducing more non-zero elements in the lower-bound matrices or other modifications. In the present context a detailed row- or column-wise examination of the differences would lead us to far. Only a limited comparison on the commodity level is included in the Appendix.

Table 2 contains a comparison of the four approaches on an aggregated level, i.e. on the level of the flow matrices. Four different approaches and three different flow matrices means that 18 comparisons can be performed.

Two different ways to compare the matrices are summarized in Table 2. One kind of comparison sums the absolute differences over all elements of two matrices compared.[14] These sums can be related to the sum of the respective matrix. Another kind of comparison looks at the differences in the row sums of the Leontief inverse matrices, that are formed with the help of the matrices to be compared. This comparison should reveal the economic relevance of differences. For the import flow matrix this kind of comparison is not available though.

The analysis presented in Table 2 shows that in most cases the differences are relatively small. Only when approach D is involved in the comparison the differences exceed 1.2 percent of the respective flow matrix. Otherwise the largest difference in terms of percentage is 0.36 of the respective matrix. The comparison based on the sum of absolute differences and on the mean absolute differences of final demand multipliers is closely aligned. Since the largest mean difference between final demand multipliers that appears in the Table 2 is 0.0036 one may conclude that there is only little economically relevant difference between the four approaches. Approaches A, B and C have relatively small differences between each

[14]It should be noted that in a certain way this implies double counting. Since the sums of the matrices compared (or, to be more precise, of each row compared) must be the same a difference in one cell must have balancing differences in other cells.

of them.

Approach B has the smallest differences to A and C. It is not clear weather approach B or C is nearer to approach D, since there is no clear way of aggregating the differences for the different flow matrices considered. However, even when neglecting the differences in the Total Flow matrix (0 for the comparison C vs. D), the sum of the sums of the differences for the other two matrices is smaller in the comparison C vs. D. Thus approach C rather than approach B seems to be nearest to approach D.

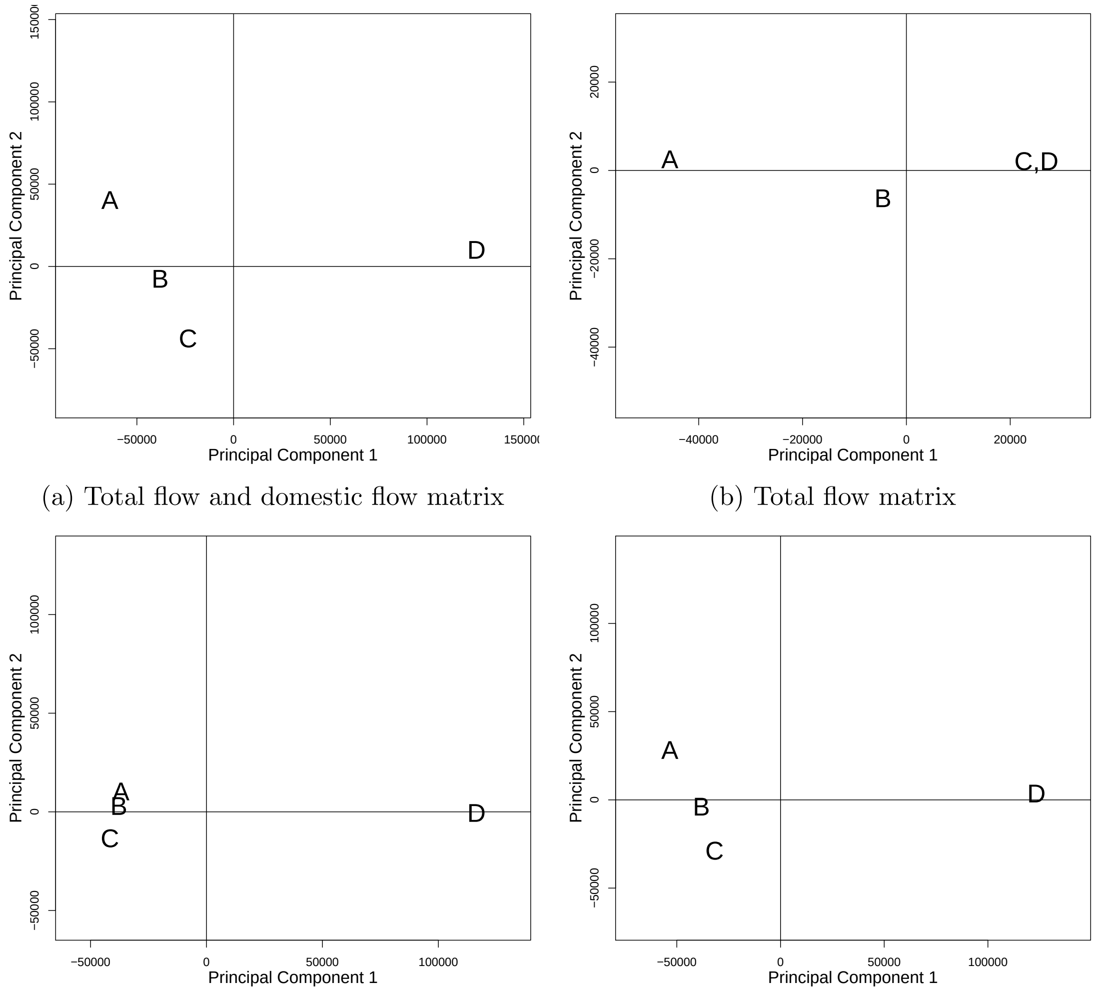
In addition to the analysis summarized in Table 2, we perform a principal component analysis (PCA) to compare the four approaches and calculate the position of each approach in the two-dimensional space spanned by the first two principal components. The data set for this PCA contains four observations, corresponding to the approaches A, B, C and D. The variables are the elements of the flows matrices to be compared. We consider only those flows, where the difference between the minimum and the maximum is greater than 100,000 EUR, leaving us with 255 variables for the Total Flow matrix, 782 variables for the Import-Flow matrix and 864 variables for the Domestic Flow matrix. Additionally, we perform a PCA for the merged data set based on the Total Flow matrix and the Domestic Flow matrix, thus containing 1119 variables.[15] The results of these four PCAs are displayed in Figure 1.

While plot (a) in Figure 1 summarizes the overall differences between the four approaches, the other three plots only provide something like a decomposition, thus helping to interpret plot a). Therefore we only discuss plot (a). Note also that in the plots we used identical ranges on the x- and y-axis in order to communicate that principal component 1 is much more important than principal component 2 in terms of variance.

The PCA shows that principal component 1 captures primarily the differences between approach D on the one hand and the other three approaches on the other hand, i.e. differences between CSIPA- and ISIPA-based approaches. It is not easy to say what kind of effect the second principal component picks up. On a first view it seems to correspond to differences between bottom-up and top-down approaches, but this is not entirely true, since approach D, which is a top-down approach, is situated in the middle with respect to principal component 2.

The PCA corroborates the impression gained from Table 2. Approach B is situated between approaches A and C, while approach D is far away. In terms of euclidean distance based on the first two principal components approach C is slightly nearer to approach D than is approach C. With this exception approach B is situated nearest to all others.

[15] It is not necessary to merge the data sets based on all three flow matrices, since the Import-Flow matrix is linearly dependent on the other two matrixes. For the same reason a PCA on the merged dataset based on the Total Flow matrix and the Import-Flow matrix or a PCA on the merged dataset based on the Domestic Flow matrix and the Import-Flow matrix produces the same results as the ones for the merged dataset chosen by us.



(a) Total flow and domestic flow matrix

(b) Total flow matrix

(c) Import flow matrix

(d) Domestic flow matrix

Figure 1: Principal Component Analysis (PCA) for the comparison of four approaches for the construction of a consistent set of flow matrices

4.5 Construction of commodity-by-component Value Added and commodity-by-categories Employment matrices

The construction of component-by-commodity Value Added and category-by-commodity Employment matrices involves problems similar to the ones discussed in the previous subsection. A problem with negative elements must be avoided and the matrices have to be consistent within themselves. Again, the Enhanced Almon's algorithm offers a practicable way to ensure these properties.

The raw tables for Value Added are in dimension component-by-activity. The six components are:

Wages and salaries

Employers' social contributions
Other taxes on production
Other subsidies on production
Consumption of fixed capital
Operating surplus, net

Among these components, *Other subsidies on production* is non-positive and *Operating surplus, net* contains negative values. *Operating surplus, net*, as a residual, requires special treatment for theoretical reasons. As an intermediate step we formed a modified Value Added matrix, in which we replaced *Other subsidies on production* by its absolute values and *Operating surplus, net* by *Operating surplus, net + Consumption of fixed capital.*

Using the formula analogous to equation (1),

$$\mathbf{W}_c = \mathbf{W}_a(\mathbf{M}')^{-1}, \tag{17}$$

where $\mathbf{W}_c$ is the modified Value Added matrix in dimension component-by-commodity and $\mathbf{W}_a$ is the modified Value Added matrix in dimension component-by-activity, we analyse the relevance of negative elements in $\mathbf{W}_c$.

There are only 3 elements in $\mathbf{W}_c$, in component *Other subsidies on production.* The sum of these negative elements is -9,2 Mill. EUR or -0.27 percent of the sum of *Other subsidies on production.*

To calculate $\mathbf{W}_c$ without negatives, we used the Enhanced Almon's algorithm, then performed the modifications backward, i.e. changing the sign of *Other subsidies on production* back to negative again and calculating *Operating surplus, net* as difference.

The final version of the component-by-commodity Value Added matrix must fulfil the IO balance equation:

$$\mathbf{x}' = \mathbf{e}'\mathbf{R} + \mathbf{e}'\mathbf{W}_c, \tag{18}$$

where $\mathbf{e}$ is a summation vector. To ensure this restriction we applied RAS to $\mathbf{W}_c$. An alternative procedure would have been to apply RAS to the matrix $[\mathbf{R}'\ \mathbf{W}_c']'$, i.e. the appropriately stacked matrix.

The raw tables for Employment are in dimension activity-by-category. The six categories are:

Jobs Self-employed persons
Jobs Employees
Jobs Total
Full-time equivalence Self-employed persons
Full-time equivalence Employees
Full-time equivalence Total

At calculating the category-by-commodity Employment matrix we have to take account of the restriction that *Totals* must equal *Self-employed persons* + *Employees* and that *Full-time equivalences (FTE)* must be smaller than *Jobs*. This was ensured by the following procedure:

(i) calculate commodity-based *FTE Self-employed persons* and *FTE Employees* by using the Enhanced Almon's algorithm,

(ii) calculate commodity-based *Jobs Self-employed persons* and *Jobs Employees* by using the Enhanced Almon's algorithm and supplying commodity-based *FTE Self-employed persons* and *FTE Employees* as lower-bounds,

(iii) calculate *Totals* as sums of *Self-employed persons* and *Employees*.

The close inspection of the component-by-commodity Value Added and category-by-commodity Employment matrices offers many ways to check for plausibility. In an early phase of the data preparation we detected large differences between *Wages and salaries* per *Jobs Employees* when comparing commodity-based and activity-based values. Such implausible results induced us to return to the data preparation phase.

5 Conclusions

In this paper we presented methods and procedures for the construction of a consistent set of input-output tables. It was shown that many restrictions must be taken account of. Among these the most prominent is certainly the avoidance of negative elements, but other aspects can pose even harder to solve problems.

In an application to Austria the practicability of the presented methods and procedures was demonstrated. This application was done within the framework of the new Austrian INFORUM model.

The presented methods and procedures form only a part of an iterative process. When problems turn up at a later phase of the task, it may be necessary to return to the data preparation phase. The appearance of negative elements may indicate deeper problems in the data. Other plausibility checks are essential, too. The process of constructing a consistent set of input-output tables requires much effort on a detailed level.

Particular emphasis was put on the problem of constructing a consistent set of Total Flow, Import-Flow and Domestic-Flow matrices. It was mainly the complexity of this task that motivated us to further develop Almon's algorithm. The Generalized and Enhanced Almon's algorithm presented in section 2 is a valuable tool for the "automatic" compliance with lower bound restrictions. We analysed four different approaches, one bottom-up, two different top-down approaches and a difference-based approach. From a theoretical perspective all of these approaches are correct. They are all in accordance with the Commodity Technology Assumption. In the application the four approaches did not yield very different Total Flow, Import-Flow and Domestic-Flow matrices in terms of economic relevance.

We do not know the "true" flow matrices. Therefore there is no way to say which approach is the best. We presented a comparison that demonstrated how far from each other the different approaches are. This may guide the practicionner in his or her decision what approach to use. It is also possible to mix these approaches and the mixture-ratios can be varied for each sector. Taken as a whole we think that the repertoire we have developed in this paper offers additional convenience at the task of compiling a consistent set input-output matrices.

References

Almon, C. (2000). Product-to-product tables via product-technology with non negative flows. *Economic Systems Research*, **12**(1), 27–43.

Armstrong, A. G. (1975). Technology assumptions in the construction of U.K. input-output tables. In Allen, R. I. G. and Gossling, W. F. (eds.), *Estimating and Projecting Input-Output Coefficients*, pp. 68–93. Input-Output Publishing Company.

Böhm, B. and Richter, J. (2006). AEIOU – towards a new Austrian INFORUM model. Paper for the 14th International INFORUM Conference, Traunkirchen, Austria, September 2006.

Bohlin, L. and Widell, L. M. (2006). Estimation of commodity-by-commodity matrices. *Economic Systems Research*, **18**(2), 205–215.

Kop Jansen, P. and ten Raa, T. (1990). The choice of model in the construction of input-output coefficients matrices. *International Economic Review*, **31**(1), 213–227.

Parve, R. (2004). Derivation of product-by-product IO matrices using PTP and treatment of secondary production. Paper for the 12th International INFORUM Conference, Ascea, Italy, September 2004.

ten Raa, T. and Rueda-Cantuche, J. M. (2003). The construction of input-output coefficients matrices in an axiomatic context: Some further considerations. *Economic Systems Research*, **15**(4), 439–455.

Statistik Austria (2005). *Aufkommens- und Verwendungstabelle*. Vienna: Statistik Austria.

6 Appendix

Table 3: Classification of commodities and used abbreviations

Abbr.	CPA-Code	Description
AgricF	01+05	Products of agriculture and fishing
Forest	02	Products of forestry
CoalLP	10	Coal and lignite; peat
CruOre	11	Crude petroleum, natural gas, metal ores
MinQua	14	Other mining and quarrying products
FoodBe	15	Food products and beverages
Tobacc	16	Tobacco products
Textil	17	Textiles
Appar	18	Wearing apparel; furs
Leathe	19	Leather and leather products
Wood	10	Wood and products of wood
Paper	21	Pulp, paper and paper products
PrintM	22	Printed matter and recorded media
RefPet	23	Coke, refined petroleum products
Chem	24	Chemicals, chemical products
RubbPl	25	Rubber and plastic products
GlassC	26	Other non-metallic mineral products
BasMet	27	Basic metals
MetPrd	28	Fabricated metal products
MachEq	29	Machinery and equipment n.e.c.
OfMach	30	Office machinery and computers
ElecMA	31	Electrical machinery and apparatus
RadCEq	32	Radio, TV and communication equipment
MedIns	33	Med., precision, opt. instruments; watches, clocks
MotVeh	34	Motor vehicles, trailers and semi-trailers
OthTra	35	Other transport equipment
FurOth	36	Furniture; other manufactured goods n.e.c.
Recov	37	Recovered secondary raw materials
ElecD	40.1	Electrical energy
GasSHD	40.2+40.3	Gas, steam and hot water
WaterD	41	Water; distribution services of water
Constr	45	Construction work
TRMotV	50	Trade and repair services of motor vehicles etc.
Wholes	51	Wholesale and comm. trade serv., ex. of motor vehicles
Retail	52	Retail trade serv., repair serv., exept of motor vehicles
HotRes	55	Hotel and restaurant services
TransW	60+61	Land and water transport and transport via pipeline services
AirTra	62	Air transport services
SeTra	63	Supporting transport services; travel agency services
SeTele	64	Post and telecommunication services
SeFIM	65	Financial intermediation services (ex. insurance serv.)
SeInsu	66	Insurance and pension funding services
SeAFIM	67	Services auxiliary to financial intermediation
SeReal	70	Real estate services
SeRent	71	Renting services of machinery and equipment
SeComp	72	Computer and related services
RnD	73	Research and development services
SeBus	74	Other business services
SePubA	75	Public administration services etc.
EduSer	80	Education services
SeHeal	85	Health and social work services
Sewage	90	Sewage and refuse disposal services etc.
SeOrga	91	Membership organisation services n.e.c.
SeCuSp	92	Recreational, cultural and sporting services
SeOth	93	Other services
SePrHH	95	Private households with employed persons

Table 4: Comparisons A vs. B, A vs. B and B vs. C for selected product groups

CPA	Product Group	A vs. B	A vs. C	B vs. C
Total Flow matrix:				
01+05	AgricF	16112 (0.34)	20982 (0.45)	4881 (0.10)
21	Paper	20070 (0.54)	38966 (1.04)	19316 (0.52)
26	GlassC	5371 (0.14)	9178 (0.25)	4093 (0.11)
27	BasMet	43807 (0.70)	80903 (1.29)	37646 (0.60)
40.1	ElecD	14390 (0.27)	24508 (0.46)	12157 (0.23)
45	Constr	46395 (0.53)	70766 (0.81)	35499 (0.41)
51	Wholes	14086 (0.14)	22844 (0.23)	11197 (0.11)
52	Retail	8180 (0.45)	15560 (0.86)	7763 (0.43)
60	TransW	5090 (0.09)	8451 (0.16)	4224 (0.08)
62	AirTra	17700 (0.93)	34252 (1.80)	17129 (0.90)
71	SeRent	5164 (0.14)	8849 (0.23)	4424 (0.12)
92	SeCuSp	35840 (1.8)	67408 (3.39)	33704 (1.69)
Import-Flow matrix:				
01+05	AgricF	9985 (1.15)	14047 (1.62)	4508 (0.52)
11	CruOre	608 (0.02)	19504 (0.68)	19586 (0.68)
10	Wood	803 (0.10)	3042 (0.37)	3603 (0.44)
21	Paper	773 (0.04)	13866 (0.71)	13307 (0.69)
27	BasMet	18533 (0.43)	31648 (0.74)	17364 (0.40)
29	MachEq	1082 (0.03)	4965 (0.12)	5393 (0.13)
32	RadCEq	1893 (0.08)	6002 (0.27)	5064 (0.23)
34	MotVeh	877 (0.02)	3943 (0.08)	3070 (0.06)
36	FurOth	1228 (0.18)	3381 (0.50)	2568 (0.38)
51	Wholes	3992 (0.46)	8835 (1.02)	6563 (0.76)
62	AirTra	1415 (0.20)	7203 (1.01)	6268 (0.88)
92	SeCuSp	6473 (0.78)	19526 (2.35)	13053 (1.57)
Domestic Flow matrix:				
01+05	AgricF	6139 (0.16)	9007 (0.24)	2890 (0.08)
11	CruOre	24 (0.01)	19668 (5.03)	19672 (5.04)
21	Paper	19483 (1.08)	37465 (2.08)	18733 (1.04)
26	GlassC	4589 (0.18)	9767 (0.39)	5345 (0.21)
27	BasMet	27739 (1.42)	54598 (2.80)	27444 (1.41)
40.1	ElecD	14390 (0.32)	24508 (0.55)	12123 (0.27)
45	Constr	46395 (0.57)	70766 (0.87)	35385 (0.43)
51	Wholes	11422 (0.13)	22014 (0.25)	11007 (0.12)
52	Retail	8180 (0.47)	15560 (0.89)	7762 (0.44)
62	AirTra	17126 (1.44)	33659 (2.84)	16829 (1.42)
71	SeRent	4425 (0.13)	8845 (0.25)	4422 (0.13)
92	SeCuSp	33704 (2.91)	67408 (5.83)	33721 (2.92)

For every type of flow matrix only those product groups are included in the table where at least one of the comparisons made yields a value greater than a specific value which is chosen such that exacly 12 product groups are included. The values are the sums of absolute differences between corresponding rows of the matrices resulting from the approaches to be compared. Sums are in 1000 EUR. Percentages are in parentheses.

Table 5: Comparisons A vs. D, B vs. D and C vs. D for selected product groups

CPA	Product Group	A vs. D		B vs. D		C vs. D	
Total Flow matrix:							
01+05	AgricF	20982	(0.45)	4881	(0.10)	0	(0)
21	Paper	38966	(1.04)	19316	(0.52)	0	(0)
26	GlassC	9178	(0.25)	4093	(0.11)	0	(0)
27	BasMet	80903	(1.29)	37646	(0.60)	0	(0)
40.1	ElecD	24508	(0.46)	12157	(0.23)	0	(0)
45	Constr	70766	(0.81)	35499	(0.41)	0	(0)
51	Wholes	22844	(0.23)	11197	(0.11)	0	(0)
52	Retail	15560	(0.86)	7763	(0.43)	0	(0)
60+61	TransW	8451	(0.16)	4224	(0.08)	0	(0)
62	AirTra	34252	(1.80)	17129	(0.90)	0	(0)
71	SeRent	8849	(0.23)	4424	(0.12)	0	(0)
92	SeCuSp	67408	(3.39)	33704	(1.69)	0	(0)
Import-Flow matrix:							
01+05	AgricF	51511	(5.96)	41744	(4.83)	39663	(4.59)
21	Paper	42253	(2.18)	42976	(2.21)	49927	(2.57)
24	Chem	93871	(1.59)	93754	(1.59)	94118	(1.60)
25	RubbPl	61309	(2.53)	61252	(2.52)	61229	(2.52)
27	BasMet	76793	(1.78)	73567	(1.71)	70773	(1.64)
28	MetPrd	107310	(4.35)	107327	(4.35)	107356	(4.35)
29	MachEq	152202	(3.76)	153245	(3.78)	156394	(3.86)
31	ElecMA	88920	(3.09)	88920	(3.09)	88920	(3.09)
51	Wholes	143301	(16.57)	147179	(17.02)	151427	(17.51)
55	HotRes	125045	(11.18)	125065	(11.18)	126034	(11.26)
62	AirTra	43622	(6.10)	44597	(6.24)	49964	(6.99)
74	SeBus	106177	(4.87)	106858	(4.90)	107561	(4.94)
Domestic Flow matrix:							
21	Paper	80624	(4.48)	61439	(3.42)	49927	(2.78)
24	Chem	94832	(12.48)	94197	(12.40)	94118	(12.39)
27	BasMet	109447	(5.61)	88658	(4.55)	70773	(3.63)
28	MetPrd	107497	(2.66)	107404	(2.66)	107356	(2.66)
29	MachEq	154302	(9.32)	153252	(9.26)	156394	(9.45)
31	ElecMA	88920	(7.88)	88920	(7.88)	88920	(7.88)
45	Constr	76973	(0.94)	38452	(0.47)	6220	(0.08)
51	Wholes	163296	(1.82)	153271	(1.71)	151427	(1.69)
55	HotRes	129200	(7.45)	127123	(7.33)	126034	(7.27)
62	AirTra	76594	(6.46)	60079	(5.07)	49964	(4.21)
74	SeBus	110144	(0.84)	108160	(0.82)	107561	(0.82)
92	SeCuSp	78946	(6.82)	51371	(4.44)	43410	(3.75)

For every type of flow matrix only those product groups are included in the table where at least one of the comparisons made yields a value greater than a specific value which is chosen such that exacly 12 product groups are included. The values are the sums of absolute differences between corresponding rows of the matrices resulting from the approaches to be compared. Sums are in 1000 EUR. Percentages are in parentheses.