Disclosure of Product System Models in Life Cycle Assessment: Achieving Transparency and Privacy

Brandon Kuczenski

Institute for Social, Behavioral, and Economic Research
University of California, Santa Barbara

2018-04-23

Supporting Information

1 List of Mathematical Symbols

The symbols used in the text are defined in Table S1.

2 Mathematical Relations

2.1 Final Demand Augmentation

This is a proof of Equation 2 in the manuscript.

Given a life cycle inventory database (LCIDB) containing input-output matrix A and emission matrix B, and a final demand vector $\mathbf{y}$, construct an augmented inventory database $\tilde{A} = \begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{y} & A \end{bmatrix}$, $\tilde{B} = [\mathbf{0}, B]$, and canonical functional unit $\tilde{\mathbf{y}} = [1, 0, 0, \ldots, 0]^T$. Show that

$$B \cdot (I - A)^{-1} \cdot \mathbf{y} = \tilde{B} \cdot (I - \tilde{A})^{-1} \cdot \tilde{\mathbf{y}} \tag{1}$$

We begin by constructing $(I - \tilde{A})^{-1}$. Because the matrix is block triangular, we only need to determine the value of $\mathbf{q}$ in the matrix below:

$$I = (I - \tilde{A})^{-1} \cdot (I - \tilde{A}) \tag{2}$$

$$\begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & I \end{bmatrix} = \begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{q} & (I - A)^{-1} \end{bmatrix} \cdot \begin{bmatrix} 1 & \mathbf{0}^T \\ -\mathbf{y} & (I - A) \end{bmatrix} \tag{3}$$

Table S1: List of Mathematical Symbols

Symbol	Formula	Size	Description	ref.
Dimensions				
m		scalar	Number of emissions	Eq. 1
n		scalar	Number of process outputs in background database	Eq. 1
p		scalar	Number of nodes in foreground model	Fig. 2
p'		scalar	Number of cut-off nodes in foreground	Fig. 4c
N	$n + p$	scalar	size of augmented matrix	n.a.
Classical LCA				
A		$n \times n$	Direct requirements matrix (Leontief)	Eq. 1
$I - A$		$n \times n$	Technology matrix	"
B		$m \times n$	Emission matrix	"
$\mathbf{y}$		$n \times 1$	Final demand vector	"
$\mathbf{x}$	$(I - A)^{-1} \cdot \mathbf{y}$	$m \times 1$	Activity level vector	Fig. 1
$\mathbf{b}$	$B \cdot \mathbf{x}$	$m \times 1$	Life cycle inventory	"
$\mathbf{c}$		$m \times 1$	Characterization vector	Eq. 1
s	$\mathbf{c}^T \cdot \mathbf{b}$	scalar	LCIA Result	"
Product System Model				
$\tilde{\mathbf{y}}$	$[1, 0, 0, \ldots, 0]^T$	$N \times 1$	Canonical functional unit vector	Eq. 2
$\tilde{A}$		$N \times N$	Augmented A matrix	Eq. 3
$\tilde{B}$		$m \times N$	Augmented B matrix	"
A_f		$p \times p$	Foreground matrix	"
A_d		$n \times p$	Background dependency matrix	"
B_f		$m \times p$	Foreground emission matrix	"
B_x	$B \cdot (I - A)^{-1}$	$m \times n$	Background LCI database matrix	Eq. 4
$\tilde{\mathbf{y}}_f$	$[1, 0, 0, \ldots, 0]^T$	$p \times 1$	Canonical foreground functional unit	"
$\tilde{\mathbf{x}}$	$(I - A_f)^{-1} \cdot \tilde{\mathbf{y}}_f$	$p \times 1$	Foreground activity vector	Fig. 2
Foreground Aggregation				
$\tilde{\mathbf{b}}_f$	$B_f \cdot \tilde{\mathbf{x}}$	$m \times 1$	Aggregated foreground emissions	Eq. 5
$\tilde{\mathbf{a}}_d$	$A_d \cdot \tilde{\mathbf{x}}$	$n \times 1$	Aggregated dependencies	"
$\tilde{\mathbf{b}}_x$	$B_x \cdot \tilde{\mathbf{a}}_d$	$m \times 1$	Aggregated background	"
$\tilde{\mathbf{a}}_f$		$p' \times 1$	Vector of foreground cut-off flows	Fig. 4c
$\check{\tilde{\mathbf{b}}}$	$\tilde{\mathbf{b}}_x + \tilde{\mathbf{b}}_f$	$m \times 1$	Aggregated life cycle inventory	Fig. 4f
Private Aggregation				
$\tilde{\mathbf{a}}_{d,priv}$		$n \times 1$	Private dependencies	Eq. 7
$\tilde{\mathbf{a}}'_d$	$\tilde{\mathbf{a}}_d - \tilde{\mathbf{a}}_{d,priv}$	$n \times 1$	Disclosed dependencies	"
$A_{d,priv}$		$n \times p$	Private dependency matrix	Eq. 8
$A_{d,pub}$	$A_d - A_{d,priv}$	$n \times p$	Public dependency matrix	"
$B_{b,priv}$		$m \times p$	Private foreground emission matrix	"
$B_{f,pub}$	$B_f - B_{f,priv}$	$m \times p$	Public foreground emission matrix	"
$\mathbf{b}_{priv}$	$(B_{f,priv} + B_x \cdot A_{d,priv}) \cdot \tilde{\mathbf{x}}$	$m \times 1$	Private life cycle inventory	Eq. 9
φ	Eq. 12	scalar	Disclosure completeness metric	Eq. 12

It can be seen that $\mathbf{q} - (I - A)^{-1} \cdot \mathbf{y} = \mathbf{0}$ is a necessary condition to satisfy the equality, and thus $\mathbf{q} = (I - A)^{-1} \cdot \mathbf{y}$.

We know that $\tilde{\mathbf{y}}$ selects the first column of its argument, so:

$$(I - \tilde{A})^{-1} \cdot \tilde{\mathbf{y}} = \begin{bmatrix} 1 \\ \mathbf{q} \end{bmatrix} \tag{4}$$

And therefore

$$\tilde{B} \cdot (I - \tilde{A})^{-1} \cdot \tilde{\mathbf{y}} = \begin{bmatrix} \mathbf{0} & B \end{bmatrix} \cdot \begin{bmatrix} 1 \\ \mathbf{q} \end{bmatrix} \tag{5}$$

$$= B \cdot \mathbf{q} \tag{6}$$

$$= B \cdot (I - A)^{-1} \cdot \mathbf{y} \tag{7}$$

thus completing the proof.

2.2 The Study Foreground Equation

This section contains the derivation of Equation 4 in the manuscript, in two steps.

2.2.1 "Flattening" the background database

First, information about the background database is removed from $\tilde{A}$ and replaced by an aggregated LCI database in a process I call "flattening."

Given a canonical LCA foreground study, show that the following formulations are equivalent:

$$\tilde{A} = \begin{bmatrix} A_f & 0 \\ A_d & A \end{bmatrix}; \quad \tilde{B} = \begin{bmatrix} B_f & B \end{bmatrix} \tag{8}$$

$$\tilde{A}_{flat} = \begin{bmatrix} A_f & 0 \\ A_d & 0 \end{bmatrix}; \quad \tilde{B}_{flat} = \begin{bmatrix} B_f & B_x \end{bmatrix} \tag{9}$$

where $B_x = B \cdot (I - A)^{-1}$.

It is sufficient to show that $\tilde{B} \cdot (I - \tilde{A})^{-1} = \tilde{B}_{flat} \cdot (I - \tilde{A}_{flat})^{-1}$.

Using the shorthand $Z = I - A$, we define the following identity:

$$I* = \begin{bmatrix} I & 0 \\ 0 & Z \end{bmatrix}^{-1} \cdot \begin{bmatrix} I & 0 \\ 0 & Z \end{bmatrix} \tag{10}$$

We introduce the identity in between the two terms of the LCIDB:

$$\tilde{B} \cdot (I - \tilde{A})^{-1} = \begin{bmatrix} B_f & B \end{bmatrix} \cdot I * \cdot \begin{bmatrix} I - A_f & 0 \\ -A_d & I - A \end{bmatrix}^{-1} \tag{11}$$

$$= \begin{bmatrix} B_f & B \end{bmatrix} \cdot \begin{bmatrix} I & 0 \\ 0 & Z \end{bmatrix}^{-1} \cdot \begin{bmatrix} I & 0 \\ 0 & Z \end{bmatrix} \cdot \begin{bmatrix} I - A_f & 0 \\ -A_d & Z \end{bmatrix}^{-1} \tag{12}$$

Using the associative property of matrix multiplication, it can be shown that $M \cdot N^{-1} = (N \cdot M^{-1})^{-1}$ for any invertible M and N of the same rank, so:

$$\tilde{B} \cdot (I - \tilde{A})^{-1} = \begin{bmatrix} B_f & B \end{bmatrix} \cdot \begin{bmatrix} I & 0 \\ 0 & Z \end{bmatrix}^{-1} \cdot \left(\begin{bmatrix} I - A_f & 0 \\ -A_d & Z \end{bmatrix} \cdot \begin{bmatrix} I & 0 \\ 0 & Z \end{bmatrix}^{-1} \right)^{-1} \tag{13}$$

$$= \begin{bmatrix} B_f & B \end{bmatrix} \cdot \begin{bmatrix} I & 0 \\ 0 & Z^{-1} \end{bmatrix} \cdot \left(\begin{bmatrix} I - A_f & 0 \\ -A_d & Z \end{bmatrix} \cdot \begin{bmatrix} I & 0 \\ 0 & Z^{-1} \end{bmatrix} \right)^{-1} \tag{14}$$

$$= \begin{bmatrix} B_f & B \cdot Z^{-1} \end{bmatrix} \cdot \left(\begin{bmatrix} I - A_f & 0 \\ -A_d & I \end{bmatrix} \right)^{-1} \tag{15}$$

But these terms are equivalent to the flattened LCIDB:

$$\tilde{B} \cdot (I - \tilde{A})^{-1} = \begin{bmatrix} B_f & B \cdot (I - A)^{-1} \end{bmatrix} \cdot \left(\begin{bmatrix} I - A_f & 0 \\ -A_d & I \end{bmatrix} \right)^{-1} \tag{16}$$

$$= \begin{bmatrix} B_f & B_x \end{bmatrix} \cdot \left(I - \begin{bmatrix} A_f & 0 \\ A_d & 0 \end{bmatrix} \right)^{-1} \tag{17}$$

$$= \tilde{B}_{flat} \cdot (I - \tilde{A}_{flat})^{-1} \tag{18}$$

thus completing the proof.

2.2.2   Separating Background from Foreground

Equation 4 in the manuscript is derived from Equation 31 above. It was also derived in Kuczenski (2015).

We begin by computing $(I - \tilde{A}_{flat})^{-1}$, again taking advantage of the matrix's block triangularity. Find Q such that:

$$\begin{bmatrix} (I - A_f) & 0 \\ -A_d & I \end{bmatrix} \cdot \begin{bmatrix} (I - A_f)^{-1} & 0 \\ Q & I \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \tag{19}$$

It can be seen that $Q = A_d \cdot (I - A_f)^{-1}$ is a necessary condition to satisfy the equality, and thus:

$$(I - \tilde{A})^{-1} = \begin{bmatrix} (I - A_f)^{-1} & 0 \\ A_d \cdot (I - A_f)^{-1} & I \end{bmatrix} \tag{20}$$

.

Substituting this into the LCA system equation:

$$s = \mathbf{c}^T \cdot \tilde{B}_{flat} \cdot (I - \tilde{A}_{flat})^{-1} \tilde{\mathbf{y}} \tag{21}$$

$$= \mathbf{c}^T \cdot \begin{bmatrix} B_f & B_x \end{bmatrix} \cdot \begin{bmatrix} (I - A_f)^{-1} & 0 \\ A_d \cdot (I - A_f)^{-1} & I \end{bmatrix} \cdot \begin{bmatrix} \tilde{\mathbf{y}}_f \\ \mathbf{0} \end{bmatrix} \tag{22}$$

where $\tilde{\mathbf{y}}_f$ is a canonical functional unit having the same dimension as A_f. Thus:

$$s = \mathbf{c}^T \cdot \begin{bmatrix} B_f & B_x \end{bmatrix} \cdot \begin{bmatrix} (I - A_f)^{-1} \cdot \tilde{\mathbf{y}}_f \\ A_d \cdot (I - A_f)^{-1} \cdot \tilde{\mathbf{y}}_f \end{bmatrix} \tag{23}$$

$$= \mathbf{c}^T \cdot \left(B_f \cdot (I - A_f)^{-1} \cdot \tilde{\mathbf{y}}_f + B_x \cdot A_d \cdot (I - A_f)^{-1} \cdot \tilde{\mathbf{y}}_f \right) \tag{24}$$

$$= \mathbf{c}^T \cdot \left(B_f + B_x \cdot A_d \right) \cdot (I - A_f)^{-1} \cdot \tilde{\mathbf{y}}_f \tag{25}$$

$$= \mathbf{c}^T \cdot \left(B_f + B_x \cdot A_d \right) \cdot \tilde{\mathbf{x}} \tag{26}$$

thus completing the derivation.

3   On the Equivalence of Process and Input-Output LCA

There is a level of confusion about the differing implications of process-based LCA inventory analysis as described in Heijungs and Suh (2002), and input-output analysis using a Leontief matrix. In fact, these two techniques were shown to be equivalent in Suh et al, 2010[1], whose abstract declares: "the article shows that the partitioning method in LCA is equivalent to the industry-technology model in input-output economics, and system expansion in LCA is equivalent to the by-product-technology model in input-output output economics." The article does indeed show convincingly that "the two data types can be hybridized without the loss of methodological consistency" (p.345).

Both the technology matrix A in LCA and the Leontief matrix $(I - A)$ in input output analysis are ways of encoding the relationships between the products that some processes "make" and other processes "use." In either the classical or input-output formulation, both matrices must be put into symmetric form to be invertible, and this means describing each column as having a single, reference output. This is done either through disaggregation (i.e. dividing a process by its various outputs), aggregation (i.e. combining products to form a market or industry), or some combination of these. While the task of transforming data is highly case-dependent, it is not constrained by the choice of mathematical formalism.

Suh et al, after a rich investigation of the literature, a thorough mathematical development, and a worked example, conclude that "even very large-scale LCI problems, such as those in commercial LCI databases, can be computed with a consistent mathematical framework..." that being the make-and-use framework (p. 349). In effect, the different mathematical approaches can be made equivalent. The equivalency can be made very plain by simply reinterpreting the input-output expressions *as* technology matrices. In other words, (in Suh et al) the expressions $(I - A_I)$ (eq. 6), $(I - A_C)$ (eq. 9), $(I - A_B)$ (eq. 12), and $(V' - U)$ (eq. 17) can simply be interpreted as *different*

---

[1]doi: `10.1111/j.1530-9290.2010.00235.x`; see full text for citation.

ways to formulate the process-based technology matrix. There is no inconsistency between this interpretation and the classical formulation.

Once single-output processes are obtained, one substantive difference between the various approaches is column scaling: whether the entries describe real outputs, unit outputs, or unit activity levels. The database creator must ensure that a consistent scaling is applied to both the technology matrix and the emission matrix B. The convention in LCA is for a column to represent a unit activity level. To use the input-output coefficient approach, as in the present article, the columns must be scaled to a *unit of the reference product*. This reference output is then omitted from the A matrix, and is instead supplied by the I in the expression $(I − A)$. This formulation is completely consistent with any other matrix-based formulation of LCA, including classical process LCA as traditionally practiced. In particular, the approaches described here apply equally well to LCA models using both process-based databases such as Ecoinvent, input-output databases such as Exiobase, and background databases such as the GaBi databases.

Interested readers should note that the canonical reference for LCA, ISO 14044, has nothing to say about the mathematical techniques used to solve LCA problems, and the only preference for a technology matrix over an input-output matrix is borne of tradition or custom rather than theory. SimaPro, the leading worldwide software system for scientific LCA, converted to using an iterative Leontief inverse for Version 8, as disclosed in a whitepaper published in 2014[2], realizing significant speed and memory usage improvements. OpenLCA, the open-source alternative, also reported a dramatic reduction in memory usage and computation time around the release of version 1.4 in 2014, coincident with a revised algorithm "optimized for huge databases."[3] This is highly suggestive of a switch to an iterative approach. GaBi, the more industrially oriented software, does not use a matrix formulation at all, but has always computed activity levels using iteration at the plan level.

4   Meeting the Disclosure Objectives

4.1   Transparency

The objective for transparency was identified as requiring computability, completeness, and reproducibility. These requirements can be met by a disclosure that clearly describes the identities of each row or column of A_f, A_d, and B_f containing non-zero entries, and the locations and values of those entries. In principle, a reader with this information would have the capability to construct the augmented LCIDB and perform the computation in Eq. 4. In actuality, while it is easy to reproduce a set of sparse matrices, there is considerable potential for ambiguity in stating the identities of the rows and columns of those matrices.

The foreground nodes, which make up the columns of A_f, can be chosen freely by the study author according to the objectives of the study. Large studies may contain hundreds of foreground nodes, and the nodes can correspond to physical activities, logical operations, unit conversions,

---

[2]https://www.pre-sustainability.com/news/new-calculation-engine-simapro-8
[3]http://www.openlca.org/openlca-beta-1-4-released-optimised-for-huge-databases/

accumulation or distribution points, or any other aspect of model construction that can be reflected in a process-flow diagram. The only requirement on their disclosure is that the identity of each node's reference flow, including its unit of measure, is clearly stated.

For each row in A_d containing a non-zero entry, the author must unambiguously identify the exact dataset used, including the version of the database, as well as the exact process and reference flow selected; the dimension (reference quantity or unit) of each reference flow must be specified; and the sign of the numeric entry in A_d must be consistent with the implementation of the process in the background database. Similarly, for each row in B_f containing a non-zero entry, the author must unambiguously identify the substance being exchanged with the environment, the compartment or context into which it is being exchanged, and the reference quantity or unit associated with the flow. Sign consistency must also be assured. If LCIA indicator results are included, the author must also unambiguously identify the method computed (identity of $\mathbf{c}$).

4.2   Authority and Originality

It may be observed that many LCA studies make use of data sources that have been previously published but that have not been included in any LCI reference or background database. As long as these data sources can be integrated seamlessly into an LCA computation, it is not necessary to reproduce them in a disclosure. However, in most applications, previously published data must be re-implemented by the author in the LCA software context, and often this re-implementation requires a re-interpretation of the data source as a unit process inventory, where one exchange is recognized as a reference flow and other exchange values are reported in proportionality to the reference flow.

Because of these conditions, in the vast majority of cases the LCA study disclosure must include the author's reimplementations in order to achieve both the aims for authority and originality. In so doing, the disclosure enables a critical reviewer to evaluate whether the author's implementations are generally correct and complete.

In the future, the inclusion of external data automatically can be accomplished in the same way that reference LCI data could conceivably be included automatically: by the data providers making their information available using a stable semantic reference to a specialized Web-based application programming interface (API). This would have the benefits of enabling downstream users to access the information without having to re-implement it, thus reducing the size of the disclosure necessary to describe the PSM and simplifying the task of the modeler.

5   Preparing the Foreground Matrix

The development in the main text assumes the product system model is already described as a symmetric collection of single-output processes (activities) and flows (commodities) related by coefficients. In concrete terms, this means that:

- each node in the foreground (column shared by A_f, A_d, and B_f) corresponds to a particular flow in the model's foreground (row of A_f);

- a unit activity of each node in the foreground corresponds to a unit output of each flow;

- the coefficients in A_f, A_d, and B_f are ratios expressing the quantity of the flow (row), in its native unit, exchanged per unit of the reference flow (column).

Each foreground column has both an intrinsic reference flow and a unit of measure, and the use of this flow by other processes must be made with regard to this measure. The activity level of each foreground node necessarily equals the magnitude of the total reference flow emanating from the node. For background dependencies and emissions (rows of A_d and B_f, respectively), the modeler must take care to use the units established by the references, but in the foreground case the modeler must ensure to be self-consistent in preparing the A_f matrix.

5.1   Simple Foreground Networks

Figure S1 illustrates the structure of some basic foreground configurations. Most product system models would include many of these elements mixed together. All the activities in these examples are single-output. The first (a) is a sequential model, in which each node requires one foreground input and generates one output. This model is equivalent to a "gate to gate" model. Here the weights k_i indicate the amount of the preceding reference flow that is required by the subsequent node. Figure S1(b) shows an additive model, in which the outputs of several foreground nodes are added together, equivalent to a "mixer" or a horizontal average. In this arrangement the weights should add up to a unit output of the reference node. Figure S1(c) shows a foreground model with a cyclic dependency, where some of the reference output is consumed by another foreground node.

A typical PSM may contain multiple modules or fragments that are interconnected. An example of foreground composed of several fragments is illustrated in Figure S1(d). Here, the nodes labeled 1–5 represent one fragment, which generates the foreground's canonical reference flow $\tilde{y}$. This fragment requires two interior flows from separate fragments (y_0 and y_1), and has two unconnected flows (4 and 5). The reference y_0 is supplied by a second fragment, made up of nodes 6-8. The reference y_1 is supplied by another fragment made of only one node (9). The reference flow y_1 is consumed in two different places by the other fragments.

As noted, a reviewer with access to items *d-i* and *d-iv* in the disclosure would be able to construct A_f and automatically create a process-flow diagram, annotated with information about each node.

5.2   Multi-Output Foreground Processes

The treatment of multi-functional activities in LCA is a topic of constant discussion among practitioners and theorists. While the question of the correct, appropriate, or ideal method of apportioning impacts among co-products cannot be resolved in general, the utility of a disclosure framework must be to *express* transparently how co-production was resolved within a given foreground model. Majeau-Bettez et al. (2014) recently provided a general framework (referred to hereafter
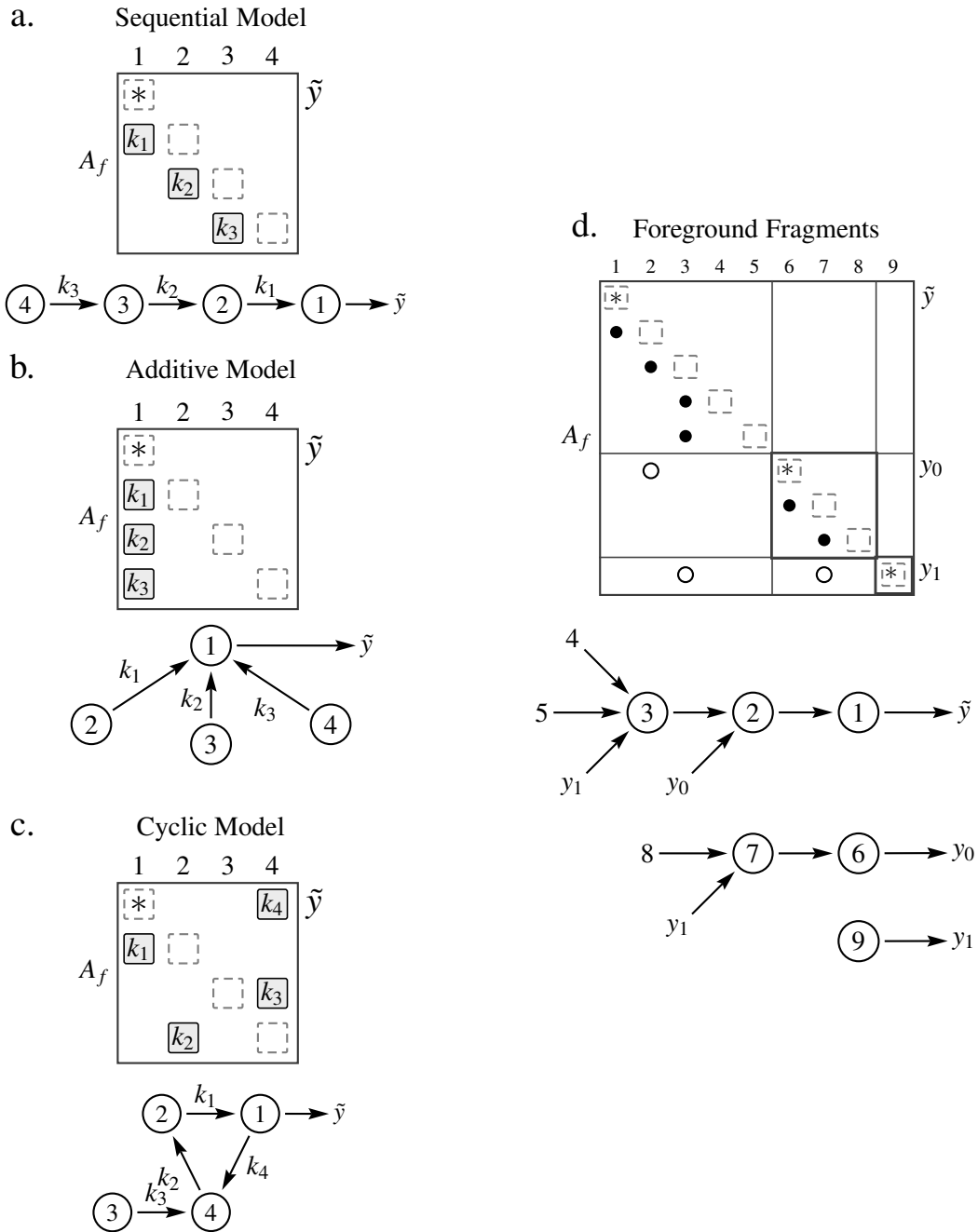
Figure S1: Equivalent matrix representations and graphs for different foregrounds.

Table S2: Example: Chlor-alkali electrolysis, membrane cell [RER] (Ecoinvent 3.2; excerpt)

	Flow	Unit	Direction	Value
Reference:				
	Chlorine, gaseous	kg	Output	1
	Hydrogen, liquid	kg	Output	0.028
	Sodium hydroxide	kg dry	Output	1.13
Requirements:				
	Electricity, medium voltage	kWh	Input	2.97
	Sodium chloride, powder	kg	Input	1.75
	Chemical factory, organics	unit	Input	4e-10
	Sludge, NaCl electrolysis	kg	Output	0.0153
Emissions:				
Chloride [16887-00-6], surface water		kg	Output	0.015
Carbon dioxide [124-38-9], urban air		kg	Output	0.0031

as MWS14) for describing the treatment of co-products, and I adopt that framework here to show how the technique used to handle co-production can be disclosed transparently for foreground processes.

5.2.1   Example System

In all strategies, an abbreviated example system drawn from Ecoinvent, reporting chlor-alkali electrolysis using a membrane cell, is used. The un-allocated system is shown in Table S2.

The objective is to disclose explicitly how the multi-output system was resolved to single-output activities for the purposes of review, reproduction, or modification. Therefore, the raw (un-allocated) process inventory is included in all cases. Examples are provided for partitioning allocation, classical system expansion, and consequential system expansion.

5.2.2   Partitioning Allocation

According to MWS14, Partitioning allocation (PA) is a "production-balanced model that artificially splits the requirements of an activity across its different outputs." The exact manner of the partitioning need not be specified, as long as the parts sum up to the whole. In the disclosure framework, any manner of production-balanced partitioning can be represented by including the raw, un-allocated inventory as one foreground column, choosing any output as the reference. The other $n − 1$ outputs are then included as allocated inventories in additional columns, for a total of n columns. Each column must be normalized to a unit reference.

Note that, although it is customary for allocated process inventories to be derived as a uniform partition of the un-allocated process based on some parameter such as mass, price or energy content, this is not necessary. For a process with n outputs the modeler has $n − 1$ degrees of freedom

Table S3: Chlor-alkali electrolysis as a foreground model, partitioning allocation

A_f	0	1	2
Chlorine, gaseous (kg)	$*$		
Hydrogen, liquid (kg)	-0.028	$*$	
Sodium hydroxide (kg dry)	-1.13		$*$
A_d	0	1	2
Electricity, medium voltage (kWh)	2.97	1.38	1.38
Sodium chloride, powder (kg)	1.75	0.811	0.811
Chemical factory, organics (unit)	4e-10	1.9e-10	1.9e-10
Sludge, NaCl electrolysis (kg)	0.0153	0.00709	0.00709
B_f	0	1	2
Chloride, to surface water (kg)	0.015	0.00695	0.00695
Carbon dioxide, to urban air (kg)	0.0031	0.00144	0.00144

in specifying the allocated inventories, with the requirement that the nth inventory equals the remainder. In the framework presented, the modeler must provide the $n-1$ allocated inventories explicitly. The reference flow for the nth process will simply be computed as the remainder when the allocated inventories are subtracted from the un-allocated inventory.

Table S3 illustrates this for the chlor-alkali example, for conventional allocation by mass. The un-altered inventory is shown in column 0, corresponding to the reference product of gaseous chlorine. Co-products are shown as additional foreground flows, with negative signs indicating that the flows are outputs. The reference for each column is indicated by an asterisk.

The inventories themselves are shown as entries in the A_d and B_f matrices. Because all co-products are already reported in mass, the allocated inventories for the two co-products are both the same and represent a scaling-down of the un-allocated inventory by the total mass of outputs (about 2.16 kg).

Using the foreground in Table S3, LCA computation using the canonical functional unit (see main text, Eq. 3) will automatically compute the self-consistent single-output result for 1 kg of chlorine gas because the allocated inventories for hydrogen and sodium hydroxide will be subtracted from the total. The foreground activity levels for the system are:

$$\tilde{\mathbf{x}} = [1, -0.028, -1.13]^T \tag{27}$$

with the resulting aggregated dependency and emission vectors being, again, the same as the normalized and allocated inventories:

$$\tilde{\mathbf{a}}_d = A_d \cdot \tilde{\mathbf{x}} = \quad [1.38, 0.811, 1.9 \cdot 10^{-10}, 0.00709]^T \tag{28}$$
$$\tilde{\mathbf{b}}_f = B_f \cdot \tilde{\mathbf{x}} = \qquad [0.00695, 0.00144]^T \tag{29}$$

At the same time, LCA computation using a functional unit of one of the co-products will simply

11

make use of the allocated inventory. In this example, all processes are identical on a mass basis, but any other partitioning allocation can also be automatically rendered self-consistent through this approach. A reviewer would be able to inspect the foreground model to determine whether the allocation was performed as described in the report.

5.2.3 "Classical" System Expansion

Although the term 'system expansion' is often used to refer to product substitution and/or alternate activity allocation, these modeling techniques are distinct and should be referred to with different names, as noted in GWS14. In the original definition of system expansion, the functional unit is simply enlarged to include all co-products. Representation of such a system is trivial and is shown in Table S4.

Table S4: Chlor-alkali electrolysis as a foreground model, classical system expansion

A_f	0	1	2
Chlorine, gaseous (kg)	$*$		
Hydrogen, liquid (kg)	-0.028	$*$	
Sodium hydroxide (kg dry)	-1.13		$*$
A_d	0	1	2
Electricity, medium voltage (kWh)	2.97		
Sodium chloride, powder (kg)	1.75		
Chemical factory, organics (unit)	4e-10		
Sludge, NaCl electrolysis (kg)	0.0153		
B_f	0	1	2
Chloride, to surface water (kg)	0.015		
Carbon dioxide, to urban air (kg)	0.0031		

In the classical system expansion case, applying the expanded functional unit of 1 kg chlorine, plus 0.028 kg hydrogen and 1.13 kg of sodium hydroxide, will result in a unit activity of the reference process, without any cut-off flows. Equivalently, the canonical functional unit, production of 1 kg of chlorine gas, will result in the same process activity level, and will also generate the co-products as cutoff flows.

5.2.4 Consequential System Expansion

The foreground disclosure framework is also well suited to describe consequential system expansion through product substitution or "alternate activity allocation," which are the terms used in MWS14. An example illustrating these methods is shown in Table S5

Here an alternate activity, namely electrolysis of water, is included for liquid hydrogen production and is specified at index 3 in the foreground (shown in blue). The inventory converts 9 kg of

Table S5: Chlor-alkali electrolysis as a foreground model, consequential system expansion. An inventory for the electrolysis of water (shown in blue) is provided as an alternate activity to produce hydrogen, while the sodium hydroxide product is assumed to substitute for for magnesium hydroxide (shown in red).

A_f	0	1	2	*3*
Chlorine, gaseous (kg)	∗			
Hydrogen, liquid (kg)	-0.028	∗		
Sodium hydroxide (kg dry)	-1.13		∗	
Hydrogen, electrolysis (kg)		*1*		∗

A_d	0	1	2	*3*
Electricity, medium voltage (kWh)	2.97			*50*
Sodium chloride, powder (kg)	1.75			
Chemical factory, organics (unit)	4e-10			
Sludge, NaCl electrolysis (kg)	0.0153			
Water, distilled (kg)				*9*
Magnesium hydroxide (kg dry)			*0.8*	

B_f	0	1	2	*3*
Chloride, to surface water (kg)	0.015			
Carbon dioxide, to urban air (kg)	0.0031			

water to 1 kg of hydrogen using 50 kWh of electricity. The hydrogen produced from the multi-output reference process is (somewhat dubiously) taken to displace hydrogen produced through the alternate activity by introducing a unity coefficient in row 3, column 1 of A_f.

Meanwhile, sodium hydroxide from the reference process is modeled to substitute the production of magnesium hydroxide through the introduction of a new row and a new nonzero coefficient in column 2 of A_d (shown in red). Here the match is inexact, and because magnesium hydroxide is more effective in some applications, 1 kg of sodium hydroxide is taken to displace only 0.8 kg of magnesium hydroxide.

6   Examples

LCI databases often contain product systems that can be modeled as foreground studies because they describe products that are not required by the background. In this section, two product systems selected from LCI databases are used to illustrate the concept of structured publication. The systems selected because are complex enough to illustrate the premise but simple enough to review easily.

Each system is illustrated as a table that shows the foreground model, cutoff flows, background dependencies and foreground emissions included in the system. Aggregation results $\tilde{\mathbf{x}}$, $\tilde{\mathbf{a}}_d$, and $\tilde{\mathbf{b}}_f$ are also reported. The table omits numeric data from most of the table for clarity. Instead, a black square indicates the presence of a nonzero value.

6.1   US LCI: Secondary Aluminum

The US LCI database contains a small background system of 39 processes, as well as 395 foreground product systems that range in size from one to 83 foreground flows.

The example product system reports the production of secondary aluminum from automotive scrap (Figure S2). The aluminum production process has direct requirements for two other foreground systems, including transportation services and quicklime production. The quicklime production itself depends on limestone extraction, which is also part of the foreground. The aluminum production requires two background systems, electricity and natural gas combustion, and reports 10 cutoff flows. The main input to the process, "Aluminum scrap, automotive," appears as a cutoff flow. Overall, the four foreground processes require the background for only combustion (five fuels), transport (three modes) and grid electricity. The same natural gas combustion and electricity models were used in all three non-transport foreground nodes.

The system also reports several direct emissions, mainly from the aluminum process. Only one resource consumption (input flow) is reported – the extraction of limestone. The direct emission modeling is limited in scope and includes a number of "unspecified" flows that may not be well characterized in many impact methods.

6.2   Ecoinvent: Organic Potatoes

Ecoinvent version 3 is provided in three different system models that reflect different linking strategies. The example is drawn from the "Allocation at the point of substitution" or APOS model of Ecoinvent 3.2, which includes 11,420 processes that produce 12,966 product flows. Of these, 10,282 are background flows and the rest are foreground flows.

The example system reports production of organic potatoes supplied to the global market (Figure S3). The foreground includes nine nodes, of which six (nodes 3 through 8) contain cyclic dependencies associated with the production of potato seeds. The Ecoinvent database's use of "markets" as mixer processes is evident in the table: node 0 is a mixer process that combines Swiss ("CH") production (node 1) weighted at 2.4% with rest-of-World ("RoW") production (node 2) weighted at 97.6%. That same market split (2.4 / 97.6) can also be seen in nodes 4 and 8.

Nodes 5 and 6, which make seeds ready "for setting," are mixed by node 8. Although nodes 5 and 6 are geographically distinguished (CH vs RoW), both can be seen to consume potato seed from the global market.

Looking at the dependency and emission lists, the different "signatures" of different kinds of processes can be seen: nodes 1, 2, 3, and 7 are clearly agricultural processes that require irrigation, tillage, manure and so on. Nodes 0 and 4 are visible as market processes, their only requirements being transport processes. Nodes 5 and 6 each consume electricity and require use of a "multi-storey building." Taken together, the CH-locality processes appear to use CH irrigation and electricity supply but are otherwise similar to their RoW counterparts. The product model includes no cutoff flows.

Aluminum, secondary, ingot, from automotive scrap, at plant [RNA]

(node) Foreground flows – A_f	0	1	2	3	
(0) Aluminum, secondary, ingot, from automotive scrap, at plant [RNA] (kg)	⬚				
(1) Aluminum recovery, transport, to plant [RNA] (kg)	1.03	⬚			
(2) Quicklime, at plant [RNA] (kg)	2.35e-05		⬚		
(3) Limestone, at mine [RNA] (kg)			1.87	⬚	
Foreground Node Weights $\tilde{x}$	1	1.03	2.35e-05	4.39e-05	
Input: CUTOFF Disposal, solid waste, unspecified, to sanitary landfill [CUTOFF Flows] (kg)	■		■		0.0842
Input: CUTOFF Filter media, at plant [CUTOFF Flows] (kg)	■				4.79e-05
Input: CUTOFF Lube oil, at plant [CUTOFF Flows] (kg)	■				8.75e-07
Input: CUTOFF Treatment gases, unspecified, at plant [CUTOFF Flows] (kg)	■				0.003
Input: CUTOFF Alloying additives, at plant [CUTOFF Flows] (kg)	■				0.021
Input: CUTOFF Chemicals, unspecified, used for wastewater treatment [CUTOFF Flows] (kg)	■				0.008
Input: CUTOFF Grain refiners, at plant [CUTOFF Flows] (kg)	■				0.0003
Input: CUTOFF Treatment salts, unspecified, at plant [CUTOFF Flows] (kg)	■				0.009
Input: CUTOFF Packaging, unspecified, at plant [CUTOFF Flows] (kg)	■				1.67e-05
Input: CUTOFF Aluminum scrap, automotive [CUTOFF Flows] (kg)	■				1.03
Background Dependencies – A_d	0	1	2	3	$\tilde{a_d}$
Transport, barge, average fuel mix [RNA] (t*km)			■		5.66e-07
Liquefied petroleum gas, combusted in industrial boiler [RNA] (l)			■		7.57e-10
Transport, combination truck, diesel powered [RNA] (t*km)		■	■		0.374
Diesel, combusted in industrial boiler [RNA] (l)			■	■	4.79e-08
Natural gas, combusted in industrial boiler [RNA] (m3)	■		■	■	0.223
Transport, train, diesel powered [RNA] (t*km)		■	■		0.0415
Electricity, at grid, US, 2000 [RNA] (kWh)	■		■	■	0.668
Bituminous coal, combusted in industrial boiler [RNA] (kg)			■	■	4.04e-06
Gasoline, combusted in equipment [RNA] (l)				■	2.25e-09
Foreground Emissions – B_f	0	1	2	3	$\tilde{b_f}$
Output: Lead [air, unspecified] (kg)	■				2.16e-07
Output: Suspended solids, unspecified [water, unspecified] (kg)	■				2.56e-05
Output: Particulates, unspecified [air, unspecified] (kg)	■		■	■	1.78e-07
Output: NMVOC, non-methane volatile organic compounds [air, unspecified] (kg)	■				4.7e-05
Output: BOD5, Biological Oxygen Demand [water, unspecified] (kg)	■				1.35e-09
Output: COD, Chemical Oxygen Demand [water, unspecified] (kg)	■				8.3e-07
Output: Sulfur dioxide [air, unspecified] (kg)			■		3.52e-09
Output: Dissolved solids [water, unspecified] (kg)	■				2.38e-08
Output: carbon dioxide [air, unspecified] (kg)			■		1.8e-05
Input: Limestone [resource, ground-] (kg)				■	4.39e-05
Output: Heavy metals, unspecified [water, unspecified] (kg)	■				2.25e-05
Output: Organic substances, unspecified [water, unspecified] (kg)	■				1.29e-06
Output: Acids, unspecified [air, unspecified] (kg)	■				4.31e-05

Figure S2: A structured product system model for secondary aluminum, drawn from US LCI. Exchange values are replaced with black squares for clarity.

market for potato, organic [GLO]

(node) Foreground flows − A_f	0	1	2	3	4	5	6	7	8	
(0) potato, organic [GLO] (kg)	■									
(1) potato, organic [CH] (kg)	0.024	■								
(2) potato, organic [RoW] (kg)	0.976		■							
(3) potato seed, organic, at farm [CH] (kg)				■	0.024					
(4) potato seed, organic, at farm [GLO] (kg)					■	1	1			
(5) potato seed, organic, for setting [RoW] (kg)						■		0.976		
(6) potato seed, organic, for setting [CH] (kg)							■	0.024		
(7) potato seed, organic, at farm [RoW] (kg)					0.976			■		
(8) potato seed, organic, for setting [GLO] (kg)		0.11	0.11	0.16				0.16	■	
Foreground Node Weights $\tilde{x}$	1	0.024	0.976	0.00315	0.131	0.128	0.00315	0.128	0.131	

Background Dependencies − A_d	0	1	2	3	4	5	6	7	8	$\tilde{a}_d$
building, multi-storey [GLO] (m3)						■	■			5.24e-06
electricity, low voltage [GLO] (kWh)						■				0.00933
potato haulm cutting [GLO] (ha)		■	■	■				■		5.04e-05
tillage, hoeing and earthing-up, potatoes [GLO] (ha)		■	■	■				■		0.000101
transport, tractor and trailer, agricultural [GLO] (metric ton*km)		■	■	■				■		0.00113
transport, freight, light commercial vehicle [GLO] (metric ton*km)	■				■					0.0258
tillage, ploughing [GLO] (ha)		■	■	■				■		5.04e-05
green manure, organic, until March [GLO] (ha)		■	■	■				■		5.04e-05
transport, freight, lorry, unspecified [GLO] (metric ton*km)	■				■					0.404
transport, freight train [GLO] (metric ton*km)	■				■					0.138
transport, freight, inland waterways, barge [GLO] (metric ton*km)	■				■					0.0857
copper oxide [GLO] (kg)		■	■	■				■		0.000107
tillage, harrowing, by spring tine harrow [GLO] (ha)		■	■	■				■		5.04e-05
potato planting [GLO] (ha)		■	■	■				■		5.04e-05
solid manure loading and spreading, by hydraulic loader and spreader [GLO] (kg)		■	■	■				■		0.717
potato grading [GLO] (kg)		■	■	■				■		1.13
tillage, harrowing, by rotary harrow [GLO] (ha)		■	■	■				■		5.04e-05
harvesting, by complete harvester, potatoes [GLO] (ha)		■	■	■				■		5.04e-05
transport, freight, sea, transoceanic ship [GLO] (metric ton*km)	■				■					0.525
liquid manure spreading, by vacuum tanker [GLO] (m3)		■	■	■				■		0.000565
electricity, low voltage [CH] (kWh)							■			0.00023
irrigation [GLO] (m3)			■					■		0.0173
tillage, currying, by weeder [GLO] (ha)		■	■	■				■		0.000101
application of plant protection product, by field sprayer [GLO] (ha)		■	■	■				■		0.000222
irrigation [CH] (m3)		■			■					0.000426

Foreground Emissions − B_f	0	1	2	3	4	5	6	7	8	$\tilde{b}_f$
Input: Occupation, construction site [natural resource, land] (m2*year)						■	■			2.1e-06
Input: Transformation, from unspecified [natural resource, land] (m2)						■	■			1.05e-06
Input: Transformation, to industrial area [natural resource, land] (m2)						■	■			1.05e-06
Input: Energy, gross calorific value, in biomass [natural resource, biotic] (MJ)		■	■	■				■		3.87
Output: Phosphate [water, ground-] (kg)		■	■	■				■		6.25e-06
Output: Cadmium, ion [water, ground-] (kg)		■	■	■				■		1.14e-09
Output: Zinc, ion [water, ground-] (kg)		■	■	■				■		1.35e-06
Output: Nitrate [water, ground-] (kg)		■	■	■				■		0.00696
… (31 rows omitted)										

Figure S3: A structured product system model for organic potato production, drawn from Ecoinvent v3.2 (APOS). Exchange values are replaced with black squares for clarity.