LCA Methodology: Data Quality

Life Cycle Analysis with Ill-Defined Data and its Application to Building Products

Jean-Luc Chevalier, Jean-François Le Téno

Centre Scientifique et Technique du Bâtiment, 24 rue Joseph Fourier, F-38400 St Martin d'Hères, France

Corresponding author: Civ. Eng. J.F. Le Téno (leteno@cstb.fr)

Abstract

In contradiction with the flow accuracy requirement of the classical LCA model, most LCA data cannot be represented by an accurate value because they loose realism in the process. It is particularly true with building products' data. Intervals are introduced to model such data, thus allowing LCA calculations to get rid off flow accuracy. Thus, interval calculation techniques for LCA are developed and the benefits from a replacement of classical LCA algorithms with these techniques are analyzed.

Key words: Data quality, LCA; ill-defined data, LCA; LCA, data quality; LCA, case studies; case studies, LCA; LCA, application to building products; building products, application to, LCA; life cycle inventory assessment; life cycle impact assessment; fuzzyness in LCA data

1 Introduction

A 1977 Environmental Protection Agency (EPA) study of the American plaster industry [13] gives data on wallboard products manufacturing. These data are now outdated (1974) and should not be used in real case LCA today. The product tree is drawn on **Fig. 1** and numerical data are listed in **Table 1**.

The upper part of **Table 1** is the inventory matrix (matrix A), which gives the amount of products (first column) flowing from process to process (first row). The bottom part is the environmental matrix (matrix B) which gives the amount of every flow taken from, or sent to, the environment (first column) by every process (first row of upper **Table 1**). Some flows, like electricity for instance, are assumed to be coming from the environment because of various allocations performed by the authors of the EPA study.

Since no global functional unit is defined at this stage, processes are not scaled one with the other. By convention, input flows are counted negatively and output flows positively. Thus Extraction of 1 ton gypsum consumes between 10 and 15kWh electricity.

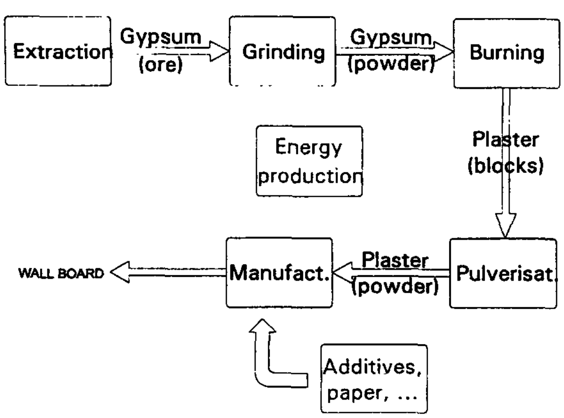


Fig. 1: Wallboard manufacturing product tree [13]

1.1 The True nature of LCA data

Those data were representative of a whole industrial branch and hence took into account the fact that most values could not realistically be known with accuracy because an LCA of a whole industrial branch has to deal with different plants, various manufacturing techniques, etc.. Most of the manufacturing process flows, though, are given very accurate values, obviously in an attempt not to give away too much proprietary information. It thus appears in that particular case that realism is ruined by accuracy.

Data the realism of which decreases as their accuracy increases are called **fuzzy** [4]. Fuzzy data are a fundamentally different type of data when compared to physical measurements, for instance. For the latter, realism is reached through accuracy.

Building products LCA data fall into the fuzzy category. Wallboard offcuts generated during installation, for instance, can differ greatly as a function of the building's geometry and the worker's skill. Also, greenhouse gas emissions from a house heating system are very sensitive to its settings and also to the inhabitants response to changes in the local climate. In both cases, flow values can change by more than one order of magnitude and cannot be modeled with realism by an accurate value.

In the classical LCA model, however, data accuracy is assumed [10] and forced by using mean values, thus deteriorating realism on the data. Any further calculation performed on this kind of data gives results in which precision hides ignorance.

Int. J. LCA 1 (2) 90-96 (1996)

Table 1: Numerical data on wallboard products manufacturing [13]

	Extraction	Grinding	Burning	Manufacturing	Pulverization
Gypsum [kg]	1000	–[1030; 1070]	0	–5	0
Gypsum powder [kg]	0	1000	–[1000; 1030]	0	0
Hemihydrate [kg]	0	0	1000	0	–1000
Wallboard [m^2]	0	0	0	10	0
Plaster [kg]	0	0	0	–850	1000
$CaCl_2$ [kg]	0	–[1; 2]	0	0	0
Dust [kg]	[0; 5]	[0; 3]	[0; 5]	0.5	[0; 0.5]
Electricity [kWh]	–[10; 15]	–[2; 5]	–[2; 5]	–[10; 20]	–[10; 20]
Fuel [kWh]	–[50; 100]	0	0	0	0
Glass Fibre [kg]	0	0	0	–2	0
Gypsum ore [kg]	–[1100; 1200]	0	0	0	0
Heat [kWh]	0	–[290.5; 406.7]	0	–348.6	0
K_2So_4 [kg]	0	0	0	–0.5	0
Lignin [kg]	0	0	0	–1	0
Paper Pulp [kg]	0	0	0	–6	0
Perlite [kg]	0	0	0	–5	0
Sawdust [kg]	0	0	0	–6	0
Soap [kg]	0	0	0	–1	0
Starch [kg]	0	0	0	–5	0
Waste [kg]	[100; 200]	0	0	0	0
Water [kg]	0	0	0	–0,6	0

Table 2: Comparison of total environmental flows computed from average and interval values

	(1) Total flows from average values	(2) Total flows from interval values[a]
$CaCl_2$ [kg]	–12.9	–[8.5; 17]
Dust [kg]	64	[5; 126]
Electricity [kWh]	–451	–[307; 697]
Fuel [kWh]	–683	–[440; 942]
Glass fibre [kg]	–20	–20
Gypsum ore [kg]	–10475	–[9685; 11301]
Heat [kWh]	–6493	–[5955; 7047]
K_2SO_4 [kg]	–5	–5
Lignin [kg]	–10	–10
Paper pulp [kg]	–60	–60
Perlite [kg]	–50	–50
Sawdust [kg]	–60	–60
Soap [kg]	–10	–10
Starch [kg]	–50	–50
Waste [kg]	1366	[880; 1884]
Water [kg]	–6	–6

[a] The calculation here was performed using techniques detailed later in this paper

As an example, the total environmental input and output flows generated when manufacturing 100 m^2 of wallboard are calculated (1) using the average of every flow and (2) using interval values. Results are given in **Table 2**.

As can been seen in case (2), some flows have more than doubled because of initial data variations in matrix A and B. Waste, for instance, varies from 0.8 ton to 1.9 ton for 100 m^2 of wallboard only. Also, while a consumption of 11 tons of ore should not be regarded as equivalent to 9.7 tons because gypsum is a non renewable resource, this variation cannot be derived from the averaged data in (1).

1.2 Two tracks to realism

Decreasing realism by averaging data for the sake of accuracy yields an unrealistic output. Realistic results should be strived for, though, and can be obtained in two different ways:

1. Realism can be "restored" afterwards by computing error bounds on results, as done by HEIJUNGS [6, 7]. But then assumptions are made that the actual data inaccuracy is not too important, so as to allow for the use of low order differential calculus. One can verify that error bounds calculated according to [7] yield inaccurate results for the wallboard example, since initial data are too inaccurate.
   It must be noted that in most publicly available LCA databases, realism is **not** restored. Finally, it is questionable whether valuable information can reasonably be derived from highly unrealistic values.
2. Another approach would be to model data fuzziness so as to keep as much realism as possible from the very beginning, and then start computations. This second approach allows the practitioner to make a real use of all information at hand.

In other words, data accuracy is not a valid assumption for LCA, and should not be strived for. Since track two is obviously more appropriate, we will now consider how fuzziness in LCA data can be modeled.

2 Ill-Defined Data Modeling for LCA

Fuzzy sets are not the only model available for fuzzy data. Other techniques exist, like intervals and probability distributions.

2.1 Intervals

An interval [1, 5, 11] can be denoted[1] by $x = [\underline{x}, \overline{x}]$. It can uniformly model accurate data ($\underline{x} = x$), data with an error ($x = [x_0 - \partial x, x_0 + \partial x]$), or even very inaccurate data ($x = [x_0 - \Delta x, x_0 + \Delta x]$). It is also the simplest model for ill-defined data since it requires very little additional information compared to a single value (two values). It is also a very familiar concept. Thus, wallboard offcuts can amount to [5; 12] %, for instance.
Calculations with intervals are possible. Note that unrealism is not explicitly modeled but still compatible with the interval model.

2.2 Fuzzy sets

Fuzzy sets are an extension of the interval concept [1, 4]. Considering that some sub-intervals are less probable (realistic) than others, fuzzy sets model the progressive shift from realism (the enclosing interval) to unrealism (a single value). Wallboard offcuts, for instance, can be modeled as in Fig. 2.

Thus, it is false to pretend that the offcut is 10 % (unrealism = 1). In a more realistic way (unrealism = 0.5), one can say that it is [7.5; 11] %. In any case (unrealism = 0), one is sure to get [5; 12] % offcuts.

Fuzzy sets clearly give more information on the data and consequently also require more information. A distribution

[1] Interval bounds are often separated with a comma. However, when bounds are figures, they will be separated with a semi-colon so as to avoid confusion with the decimal coma used in some languages.

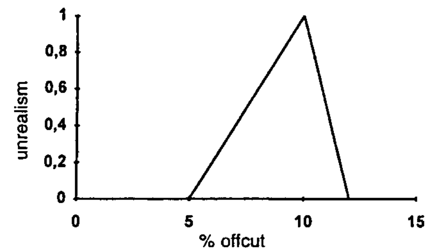


Fig. 2: Modeling wallboard offcuts flow with a fuzzy set

of intervals, called a possibility distribution [1, 4] can be derived. Calculation techniques with fuzzy sets are directly derived from their interval counterpart. Building a fuzzy set for a given data requires some thinking from the user.

2.3 Probability distributions

Contrary to fuzzy sets and the corresponding possibility distributions, probability distributions can only model accurate data [1, 4]. They also require far more information on the data than the other models.

2.4 A model for LCA data

The chosen model should be able to take into account both normal (accuracy = realism) and fuzzy data (accuracy = unrealism). It should be remembered that since LCA data is a scarce resource, a model should not ask for too much information from the user. It should, however, make full use of every piece of information available, as is obviously not the case with the single accurate value model.

In either case, probability distributions are not a good choice, although they are often automatically chosen [1]. Fuzzy sets on the contrary, perform very well as far as data fuzziness is concerned. But they are less familiar than intervals and require more information than is usually available.
Hence, we choose to model LCA data with intervals, to take advantage of their generality and ease of use. From now on we will talk about "ill-defined" rather than "fuzzy" data, since the fuzzy set model was not chosen. We will now consider how LCA calculations can be done using intervals.

3 LCA Calculations with Intervals

In order to avoid unnecessary complications, we have chosen to use HEIJUNGS' model for LCA calculations [6, 7]. Also, only life cycle inventory will be considered. Those assumptions should not limit the scope of the results presented in this paper, however.

3.1 Inventory aggregation calculations

The following is a very condensed introduction to HEIJUNGS' inventory calculations model, which is described in detail in [6]. We will assume that every necessary allocation has been performed, hence according to [6], matrix A *(see Table 1)* should be square.

Each row of matrix A is concerned with one product. Only one of those products will come out in the end (100 m^2 of wallboard, for instance). Thus, the total output must be zero for every product, except wallboard. Let α be that total product output. Knowing α and every process' product input and output from matrix A, it is possible to compute how many times p_j process j has been used by solving the following matrix equation $A.p = \alpha$.The purpose of this equation is to balance inputs and outputs.

Knowing p and every process' environmental input and output flows from matrix B, it becomes possible to calculate the total environmental flows for every process and the whole process tree leading to 100 m^2 of wallboard (β). This is simply done by computing $\beta = B.p$.

Basically, two types of operations are involved :

1. Basic operations (+, –, ÷, ×) to compute the aggregate environmental vector $\beta = B.p$.
2. Resolution of a square matrix equation $A.p = \alpha$ to calculate the occurrence p of each process.

We will now consider how those calculations can be done using intervals.

3.2 Calculating with intervals

Interval calculations have been studied since the 1960s and have been used for various problems pertaining to economic modeling [1] and process command [12], for instance. Detailed information on the mathematical issues of the **interval arithmetic** (IA) domain can be found in [11], as well as a large bibliography.

There is a new definition of arithmetic operators at the very base of IA:

$$
\begin{aligned}
&[\underline{a},\bar{a}]+[\underline{b},\bar{b}]=[\underline{a}+\underline{b},\bar{a}+\bar{b}]\\
&[\underline{a},\bar{a}]-[\underline{b},\bar{b}]=[\underline{a}-\bar{b},\bar{a}-\underline{b}]\\
&[\underline{a},\bar{a}]\times[\underline{b},\bar{b}]=[\min(\underline{a}\underline{b},\underline{a}\bar{b},\bar{a}\underline{b},\bar{a}\bar{b}),\max(\underline{a}\underline{b},\underline{a}\bar{b},\bar{a}\underline{b},\bar{a}\bar{b})]\\
&[\underline{a},\bar{a}]\div[\underline{b},\bar{b}]=[\underline{a},\bar{a}]\times[1/\bar{b},1/\underline{b}]\text{ , if }0\notin[\underline{b},\bar{b}]
\end{aligned}
\quad (1)
$$

The basic properties of these new operators are globally weaker than their real counterpart since commutativity, associativity and neutral element properties hold, but distributivity does not. Instead, one has subdistributivity :

$$a(b\pm c)\subseteq ab\pm bc$$

The main practical consequence is a possible **enlargement** of results intervals. Interval enlargement can also happen when a given variable appears more than once in an expression. Calculating $y = 1-x+x^2$ with x = [0;2], for instance, yields y = 1-[0;2]+[0;2]x[0;2] = 1-[0;2]+[0;4] = [-1;1]-[0;4] = [-1;5] instead of the expected y = [3/4;3].

Previously defined type I calculations can be performed in LCA without extra precautions using the basic definitions (1), because every variable only appears once. Interval linear systems resolution on the other hand, calls for additional developments since the algorithms used for their real counterparts rely too much on distributivity and multiple variable occurrence.

3.3 Hull of an interval equation solution

The problem is to solve a set of equations defined by the interval matrix A and right member α. A and α's coefficients are intervals. Thus one has:

$$\tilde{A}\tilde{p}=\tilde{\alpha}\quad(\tilde{A}\in A,\tilde{\alpha}\in\alpha)\qquad(2)$$

The solution of (2) is denoted by Σ (A, α). It is not exactly what could be expected from the extrapolation of the real case, as will be shown on the following example [5]:

$$\begin{cases}[2,3]p_1+[0,1]p_2=[0,120]\\ [1,2]p_1+[2,3]p_2=[60,240]\end{cases}\qquad(3)$$

In the first quadrant, one has $p_1 \geq 0$ and $p_2 \geq 0$. Hence, every p must be so that:

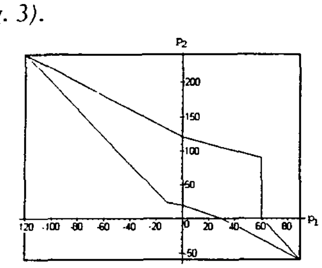
$[2p_1,3p_1+p_2]\cap[0,120]$

and $[p_1+2p_2,2p_1+3p_2]\cap[60,240]$ are not empty.

Hence, p_1 and p_2 must verify the following inequations:

$$\begin{cases}2p_1\leq 120\\ p_1+2p_2\leq 240\\ 3p_1+p_2\geq 0\\ 2p_1+3p_2\geq 60\end{cases}\qquad(4)$$

Thus in the first quadrant, solutions are enclosed in a domain whose vertex are (30,0), (60,0), (60,90), (0,120) et (0,20). The same operation must be performed for every other quadrant in order to get the equation's complete solution *(→ Fig. 3)*.



...... Limits of actual solution ——— Limits of solution's hull

Fig. 3: Solution and hull of equation (3)

Contrary to real matrix equations where the solution set is a real vector, the solution of an interval matrix equation is not an interval vector. As can be seen in **Fig. 3**, it is a complex non-convex structure. A more manageable characteristic is the hull of this structure, defined from the extreme bounds of each p_i. The hull is denoted by $\square\Sigma(A, \alpha)$. The hull of equation (3) is

$$\begin{pmatrix}[-120, 90]\\ [-60, 240]\end{pmatrix}.$$

It is drawn in **Fig. 3** with a bold line. The output of interval systems solving methods is always the hull of the solution, which will also be called «solution» in the rest of this paper.

3.4 Solving interval linear systems

Interval linear systems can be solved using an interval version of the Gauss-Seidel iteration, for instance [11]. However, depending on the initial system, the resulting solution may be too large to be useful. One of the main factors influencing the accuracy of the solution is the width of the system's coefficients. As a rule of thumb, the wider the coefficients, the less accurate (i.e. the larger) the solution will be. While some preliminary manipulations can limit this expansion [5], it cannot be completely canceled in that way.

Other techniques like Monte-Carlo sampling of the equations set can be used [12], but since every system in the sample has to be solved, they are very time consuming and precision can only be gained through a bigger sample.

COPE has shown that the hull of an equation like (2) could be calculated exactly using linear programming. This technique is discussed now.

3.5 Solving interval equations using linear programming

Linear programming's purpose is to maximize (or minimize) a function within a domain defined by constraints [3]. These constraints can be equalities or inequalities (see appendix). Both function and constraints must be linear. This is of interest to us because our equation is linear too.

What we need is the upper bound (maximum) and lower bound (minimum) of every interval p_i in vector p to get the hull of the equation's solution. Also, as can be seen from problem (3), an interval equation can be converted into a system of inequalities which is nothing but a list of constraints.

Many algorithms have been devised to solve this kind of problem[2]. They won't be discussed here but the following linear program will be solved graphically as an introduction.

$$\left|\begin{array}{l} \max z = p_1 + p_2 \\ 2p_1 \le 120 \\ p_1 + 2p_2 \le 240 \\ 3p_1 + p_2 \ge 0 \\ 2p_1 + 3p_2 \ge 60 \\ p_1 \ge 0, p_2 \ge 0 \end{array}\right. \qquad (5)$$

The constraints of (5) delineate a domain in which every point is a solution (bold line). In order to maximize z in that domain, one draws z = $p_1 + p_2$ (thin) lines for increasing values of z (maximizing) until the limit of the constrained domain is reached. As can be seen in **Fig. 4**, one then reaches z = 150. Hence the optimum solution is p = (60,90).

Now, replacing function z in (5) with z = max(p_1) and solving this new linear program, where constraints have been chosen equal to those in (4), one gets $\overline{p_1}$ in the constrained domain (i.e. 60). The same operation can be reproduced with z = min(p_1) to get $\underline{p_1}$ (i.e. 0).

The **exact** solution of the equation's solution can be computed in this way, regardless of the coefficient's width. For each $p_i = [\overline{p}_i; \underline{p}_i]$, it can be shown from [2] that both bounds can be calculated from linear programs (6a) and (6b). Both expressions are established in the appendix. Note that the constraints are the same in both programs.

[2] A good introduction to linear program solving, as well as a computer code, can be found in *Numerical Recipies*, from Cambridge University Press.

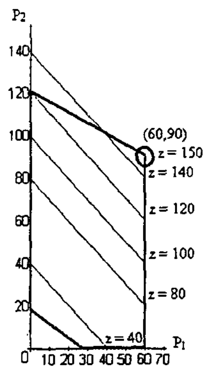


Fig. 4: Graphical resolution of linear program (5)

$$\begin{array}{ll} \underline{p_j} := \max(-p_j) & \\ \sum_{j=1}^{n} \underline{a_{ij}} p_j \le \overline{\alpha_i} & \\ \sum_{j=1}^{n} \overline{a_{ij}} p_j \ge \underline{\alpha_i} & (6a) \\ p_j \ge 0, \quad i,j \in [1..n] & \end{array} \qquad \begin{array}{ll} \overline{p_j} := \max(p_j) & \\ \sum_{j=1}^{n} \underline{a_{ij}} p_j \le \overline{\alpha_i} & \\ \sum_{j=1}^{n} \overline{a_{ij}} p_j \ge \underline{\alpha_i} & (6b) \\ p_j \ge 0, \quad i,j \in [1..n] & \end{array}$$

This technique is always applicable, but when A is an M-matrix, getting the hull can be much simpler. These matrices are of interest to us since the inventory matrixes can sometimes exhibit the same pattern. This point is discussed now.

3.6 Inventory M-matrix

An M-matrix is a special type of inverse positive matrix. A necessary condition for being an M-matrix is to be diagonal positive and negative, or zero everywhere else. A strict definition is given in the appendix.

When A is an interval M-matrix, the **exact** solution can be calculated regardless of the coefficients' width, by solving two real equations only. Taking advantage of the constant-sign nature of LCA flows (see appendix), one has [11],

$$p = [\overline{A}^{-1}\underline{\alpha}, \underline{A}^{-1}\overline{\alpha}] \qquad (7)$$

The columns of matrix A can be ordered so that each positive coefficient is on the matrix' diagonal without changing the problem. One can then expect to get an M-matrix, and thus solve the equation by using theorem (7). This should always be the case providing that there are no loops and no multiple-output processes in the product tree. However if manufacturing scraps are recycled in the manufacturing process, for instance, A will not be an M-matrix.

Interval total flows in **Table 2** were calculated using the fact that matrix A in **Table 1** is an M-matrix. We first calculated p from A.p = α with α = (0,0,100m^2,0) using theorem (7), then β^2 = B.p using IA arithmetic operations. Values of p are given in **Table 3**.

3.7 General calculation strategy

To conclude on the topic of interval system resolution in LCA, we will adopt a twofold strategy using IA for M-matrix and linear programming in the general case. In any

Table 3: Values of p for 100 m^2 of wallboard

	p
Extraction	[8.805; 9.418]
Grinding	[8.5; 8.755]
Burning	8.5
Manufacturing	10
Pulverization	8.5

case, an accurate solution is computed. These techniques are common features of most spreadsheets or could be programmed easily using macros, for instance.

4 Conclusion

"If ignorance is so wicked, and half its wickedness is our ignorance of it, then perhaps we should undertake to discover the full extent of our ignorance–to survey, catalog, and describe its range and variations, its human anatomy, if you will." [Patrick GUNKEL in *Ideonomy*]

Most LCA data cannot be modeled realistically by accurate values. Intervals have been identified as the most appropriate alternative to model the true nature of these data. Intervals are a very familiar concept for LCA practitioners and make full use of every piece of information available without requiring any additional information from the user. Intervals can also be useful to help compensate for the lack of data by replacing unavailable data with expert-defined intervals. Building product manufacturers, for instance, can have a clear idea of the approximate value of some flow, even though they it was not actually measured.

Calculation techniques necessary to cope with intervals instead of crisp values have been discussed. As a side benefit, a thorough reliability analysis is automatically performed during the aggregation process. Differential calculus expressions used for bounding errors [6] are no longer required because error bounds are given by the resulting intervals. Dominance analysis [6, 7] is straightforward using IA. Impact assessment can also be done as usual and provide interval stressors instead of real ones.

Multicriteria analysis techniques that allow for the comparison of alternative designs measured by interval criteria should now be used [14]. Also, improvement analysis, as defined by HEIJUNGS [7], will require that new expressions be established, since their current form relies on differential calculus.

This approach, while necessary for building products' LCA data, should prove useful to improve data quality for any other product. Hence, we recommend a complete replacement of the "accurate" data model by the interval model as a very powerful means to enforce openness and usability of LCA data and results.

Appendix

The set of real intervals is denoted by IR. As previously mentioned, the following properties of R still hold :

Commutativity	$a + b = b + a$ $a\,b = b\,a$
Associativity	$(a + b) \pm c = a + (b \pm c)$ $(ab)\,c = a\,(bc)$
Neutral element	$a + 0 = 0 + a = a$ $\times a = a \times 1 = a$

Apart from subdistributivity, the following weaker properties hold for IA :

$x - x \neq 0$, but $x - x \supseteq 0$
$x/x \neq$, but $x/x \supseteq 1$
$x \times x \neq x^2$, but $x \times x \supseteq x^2$

Lastly, we define the following notions :

magnitude of x $|x| = \max(|\underline{x}|, |\overline{x}|)$

center of x $\check{x} := (\overline{x} + \underline{x}) / 2$

A common representation of a linear program is max z = c.p.. z is the **objective function** to be maximized, p are the variables whose values must be found, and c denotes the respective contributions to z from an incrementation of a given variable p_j. c plays no role in our case, and will disappear from the expressions. z is subject to the following **constraints:**

$$\begin{cases} Ap = b \\ A'p \geq b' \\ A''p \leq b'' \end{cases}$$

COPE [2] has shown that the hull of a nxn linear equation can be computed by solving for 2n linear programs whose constraints and objective functions are drawn from the equation's coefficients. These constraints are :

$$\begin{cases} \sum_j \Delta a_{ij}|p_j| + \Delta b_i \geq \sum_j \check{a}_{ij}.\mathrm{sgn}(p_j).|p_j| - \check{b}_i \\ \sum_j \Delta a_{ij}|p_j| + \Delta b_i \geq -\sum_j \check{a}_{ij}.\mathrm{sgn}(p_j).|p_j| + \check{b}_i \end{cases} \quad (8)$$

with $\mathrm{sgn}(p_j) = 1$ if $p_j \geq 0$ and $\mathrm{sgn}(p^j) = -$ otherwise. Also $A = \check{A} \pm \Delta A$ and $b = \check{b} \pm \Delta b$. The objective functions are:

$$\begin{aligned} \underline{p_j} &= \max(-c_j^t.|p|) \\ \overline{p_j} &= \max(c_j^t.|p|) \end{aligned} \quad (9)$$

Since we are only looking for positive values of p, one has $\mathrm{sgn}(p_j) = 1$, $|p_j| = p_j$ and $c_i^t = (0, \ldots, 1, \ldots, 0)$. Thus (8) and (9) can be simplified and reorganized to yield (6a) and (6b).

A matrix A is an M-matrix if $A_{ij} \leq 0$ for all $i \neq j$ and $Au>0$ for some positive vector $u \in R^n$. [12]. Checking the first condition is straightforward. The second condition can be checked by solving A.u = e (e is the unit vector). If u is positive, then the second condition is met.
Every flow in LCA will be either an input flow (negative, by convention [6]), or an output flow (positive). This property holds when flows are modeled with intervals : Bounds are always both negative, or both positive. Thus flow signs are constant.

In our software implementation of the interval flow model, matrix A is LU factorized so as to make the inversion required by theorem (7) easier. The general case is solved using the Bartels-Golub version of the revised simplex algorithm [3, 8], since it can handle much more data than the tableau method with better numerical stability.

Acknowledgment

The authors would like to thank Dr. Arnold NEUMAIER for his precious help on interval analysis topics. This work is part of the "Building Products' LCA" project, started at the Centre Scientifique et Technique du Bâtiment (CSTB) in 1992, in collaboration with the Université de Savoie. It was co-financed by the Agence De l'Environnement et de la Maîtrise de l'Energie (ADEME) and CSTB.

5 References

[1] CHOOBINEH, F. and A. BEHRENS: Use of intervals and possibility distributions in economic analysis. Journal of the operational research society, 1992. 43(9):907–918

[2] COPE, J.E. and B.W. RUST: Bounds on solutions of linear systems with inaccurate data. SIAM Journal of Numerical analysis, 1979. 16(6):950–963

[3] DE WERRA, D.: Elements de programmation lineaire avec application aux graphes. 1990. Presses polytechniques romandes

[4] DUBOIS, D. and H. PRADE: Théorie des possibilités – Applications à la représentation des connaissances en informatique. Coll. Methode + Programmes. Vol. 1. 1988. Masson, Paris

[5] HANSEN, E.: Interval arithmetic in matrix computations, part I. SIAM Journal of Numerical Analysis, 1965. 2:308–20

[6] HEIJUNGS, R.: [ed.] Environmental life cycle assessment of products – Background and Guide. 1992, Leiden university

[7] HEIJUNGS, R.: A generic method for the identification of options for cleaner products. Ecological economics, 1994 (10):69–81

[8] JUILLARD, D.: Le simplexe révisé. Pascalissime, 1990 (43):34–51

[9] LE TENO, J.F. and J.L. CHEVALIER.: Etude du cycle de vie des produits de construction. Journées techniques Matériaux Energie Environnement. 1994. Sophia antipolis

[10] LE TENO, J.F. and J.L. CHEVALIER: Requirements for an LCA-based model for the evaluation and the improvement of building products environmental quality. Accepted for publication in: Building and Environment

[11] NEUMAIER, A.: Interval methods for systems of equations. Coll. Encyclopedia of mathematics and its applications, 1990. Cambridge University Press, Cambridge

[12] WANG, Z. and F. ALVARADO: Interval arithmetic in power flow systems. IEEE Transactions on power systems, 1992 7(3):1341–1349

[13] EPA: Chapter 17: The gypsum and wallboard industry. Industrial process profiles for environmental use. Vol. 17. 1977

[14] LE TENO, J.F. and MARESCHAL, B.: An interval version of Promethee for the comparison of building products' design with ill-defined data on environmental quality, To be presented to the 43rd EURO Workgroup on "Multicriteria Decision Aid", March 21st, 1996, Brest, France

LCI Data and Data Quality

Thoughts and Considerations

Bea De Smet, Mark Stalmans

Procter & Gamble European Technical Center, Temselaan 100, B-1853 Strombeek Bever, Belgium

Abstract

Life cycle inventory (LCI) is becoming an established environmental management tool that quantifies all resource usage and waste generation associated with providing specific goods or services to society. LCIs are increasingly used by industry as well as policy makers to provide a holistic 'macro' overview of the environmental profile of a good or service. This information, effectively combined with relevant information obtained from other environmental management tools, is very useful in guiding strategic environmental decision making.

LCIs are very data intensive. There is a risk that they imply a level of accuracy that does not exist. This is especially true today, because the availability of accurate LCI data is limited. Also, it is not easy for LCI users, decision-makers and other interested parties to differentiate between 'good quality' and 'poor quality' LCI data. Several data quality requirements for 'good' LCI data can be defined only in relation to the specific study in which they are used.

In this paper we show how and why the use of a common LCI database for some of the more commonly used LCI data, together with increased documentation and harmonisation of the data quality features of all LCI data, is key to the further development of LCI as a useful and pragmatic environmental management tool. Initiatives already underway to make this happen are also described.

Key words: Data quality, LCA/LCI; LCA/LCI, data quality; LCI, data quality; environmental management, LCA/LCI; databases, LCI; LCI, databases; LCI, documentation of data quality features

1 Introduction

LCI attempts to determine the overall inputs (in terms of resources including energy) and outputs (in terms of wastes) over the whole life cycle of a product or service. A typical LCI can easily incorporate thousands of individual data points. The overall quality of a LCI depends largely on the quality of the input data that are used. Due to the complex interactions of today's industrial systems, LCIs often include data from many different countries. For example, in the production of surfactants, some feedstock raw materials are produced in Malaysia or the Middle East and then further processed in Europe [1]. Additionally, data for different unit operations, and even within a unit operation or module, are usually collected from diverse sources (industry, national statistics, literature) and frequently not generated specifically for the LCI study. They may have been collected, for example, as part of an engineering study or as a regulatory parameter.

Typically, input data quality depends on the quality of the data source, the analyst's degree of knowledge of the product or process being studied, the assumptions made and the calculation and validation procedures. These factors can be used to make judgments about the quality of the specific data. However, the extent to which this can be done for each entry data point depends on the effort allocated to the