# Towards Practical Privacy-Preserving Life Cycle Assessment Computations

Cetin Sahin*, Brandon Kuczenski†, Omer Egecioglu*, Amr El Abbadi*
*Department of Computer Science
†Institute for Social, Behavioral, and Economic Research
University of California, Santa Barbara
Santa Barbara, California 93106
*{cetin, omer, amr}@cs.ucsb.edu, †bkuczenski@bren.ucsb.edu

***Abstract*—Life Cycle Assessment(LCA) is crucial for evaluating the ecological sustainability of a product or service, and the accurate evaluation of sustainability requires detailed and transparent information about industrial activities. However, such information is usually considered confidential and withheld from the public. In this paper, we present a rigorous study of privacy in the context of LCA. The main goal is to explore the privacy challenges in sustainability assessment considering the protection of trade secrets while increasing transparency of industrial activities. To overcome privacy concerns, we apply differential privacy to LCA computations considering the idiosyncratic features of LCA data. Our assessments on a specific real-life example show that it is possible to achieve privacy-preserving LCA computations without losing the utility of data completely.**

## I. Introduction

One of the greatest challenges facing global society is to ensure that the industrial goods and services required by a growing and modernizing population can be met sustainably and equitably [23]. Industrial Ecology (IE) is the study of resource requirements and the social and ecological implications of industrial activities. Its primary utility is to inform consumers, businesses, and policy makers about the magnitude and significance of *material flows* through the economy that supports specific products, technologies, or systems [26]. One primary technique in IE is life cycle assessment (LCA), a standardized methodology for estimating the total environmental implications of products or services [5], [9]. The core methodology of LCA is governed by a set of international standards [15] and is widely applied to evaluate the potential ecological consequences of consumption decisions.

Preparing an LCA requires access to a database of information about the inventory requirements and environmental emissions of industrial processes, called a life cycle inventory (LCI) database. Preparing an accurate and comprehensive LCI database is a tremendous task and the development and maintenance of these resources is an ongoing challenge [22]. Because industrial processes are typically undertaken in a competitive economic context, the operators of these processes would like to prevent potential competitors from learning sensitive information about their activities. Information that may be valuable to a competitor is often termed *confidential business information*. Inventory data about industrial processes is usually considered to be confidential, and therefore is often not available freely. This type of information is nonetheless required in order to accurately assess environmental impact. As a consequence, the historical development of LCA has long been intimately bound to questions of confidentiality [14], [10].

Despite its centrality to LCA, data privacy in the LCA domain has not formally considered. In particular, methods for privacy-preserving data publication in LCA have not been well-developed. The guiding principle behind privacy protection in LCA database preparation is that data that are regarded as secret by the owners can be concealed through aggregation with other data sets and with data sets extracted from LCI background databases (see [22], ch. 3).

In this paper, we formulate the LCA computation in a way that allows us to introduce a privacy model, and consider possible threat models and attacks that could result in an adversary learning private data. Our goal in this paper is to provide the data security community with a real sense of the challenges faced by practitioners in the field of Industrial Ecology. We explore a particular problem in LCA and explore the privacy issues and possible trade-offs between increase transparency by industrial companies and privacy protection of trade secrets that preserve competitive edge. The results of our attacks justify the concerns over publishing inventory data about industrial processes without securing with any security. To tackle this problem, we apply privacy techniques to LCA computations and illustrate their usage on a specific real life example. Our evaluations over a real life example highlight that it is possible to achieve privacy-preserving LCA publication without losing too much utility on the published data while ensuring privacy with the application of differential privacy. A straightforward optimization such as normalization, considering the idiosyncratic features of LCA data, delivers a reasonable improvement in the publication quality without sacrificing the privacy.

The followings summarize our contributions in a nutshell:

- The first formal privacy-preserving LCA computation formulation while providing more transparency.
- Verify privacy concerns of LCA practitioners by developing an attack.

- Develop a differentially private matrix multiplication that is particularly efficient in the LCA context.
- Evaluate the proposed privacy-preserving publications and propose optimization to improve publication utility.

The rest of the paper is organized as follows. The next section formulates the LCA aggregation problem and explains current practice along with privacy concerns. Section III investigates the validity of privacy concerns in LCA publications. Differentially private LCA publication techniques are presented in Section IV. The following section presents experimental evaluation. The final section concludes the paper.

## II. Formulating the LCA Aggregation Problem

### A. LCA Basics

LCA following the ISO standards describes the delivery of a product or service as a network of industrial *unit processes* whose outputs are required in order to provide a *functional unit* of utility to a user. Each unit process represents one form of industrial activity. Each edge in the network indicates a *flow* from one process to another, or between one process and the environment. Flows between processes are called *intermediate* flows, and flows between a process and the environment are called *elementary* flows. Only elementary flows may generate environmental impacts [15], [12].

LCA studies distinguish between a *foreground model*, which represents the activities under scrutiny, and a *background model*, which represents the operations of the broader economy [21]. Private data are typically contained in the foreground model. The preparation of a background database is outside the scope of an individual study, and background databases are provided and maintained by dedicated research [25] or commercial [2] organizations. Although background databases are subject to licensing restrictions, in this study they are regarded as publicly available because any party who purchases a license may inspect them freely. Background databases are assumed to be available in an aggregated form in which the relations among the different processes are not known.

An *LCA aggregation study* can be described as three sequential matrix multiplications with respect to a background database $B_x$[16]. $B_x$ is an $m \times n$ matrix that maps a set of $n$ background processes to a set of $m$ elementary flows. The foreground model is made up of a set of $p$ foreground processes, each of which is defined by its dependencies on the $n$ background processes. These are described in an $n \times p$ dependency matrix $A_d$, which comprises the study's private input data. Here $w$ is a p-element weighting vector that specifies the relative significance of the different foreground processes. The first multiplication aggregates the foreground model into a weighted dependency vector $a_p$:

$$a_p = A_d \cdot w \tag{1}$$

The dependency vector $a_p$ is then applied to the background database to determine an emission vector $b$:

$$b = B_x \cdot a_p \tag{2}$$

The vector $b$, also called a *life cycle inventory*, reports the aggregate amounts of different emissions released into the environment throughout the life cycle of the product system specified. The results of the inventory computation must be characterized with respect to a set of $t$ environmental impact categories, represented by multiplication with a $t \times m$ characterization matrix $E$.

$$s = E \cdot b \tag{3}$$

This multiplication results in a set of $t$ impact scores $s$ , which are the final results of the study. The impact scores in $s$ provide a basis to compare different product systems with equivalent functional units on the basis of their potential environmental impacts.

### B. Privacy Concerns and Current Practice

The current practice in the Industrial Ecology community is to make the result of the study $s$ (Equation 3) publicly available, so that the product system they represent can be compared to other product systems. However, it is difficult to evaluate the significance of the elements of $s$ without knowing something about $b$. For instance, an independent researcher making a critical evaluation of $s$ may wish to know whether a given environmental emission was included in $b$ with a significant value. Alternatively, a practitioner may require further knowledge about the flows in $b$, such as their geographic or temporal scope. Some research questions may require a practitioner to supply her own $E$ matrix, which is not possible if $b$ is not disclosed.

On the other hand, these requirements raise several privacy concerns over the data in $A_d$, for which $a_p$ is a proxy. In the absence of a formal understanding of the privacy implications of disclosing $b$, it is common practice in the community to withhold $b$ and only publish $s$. As mentioned earlier, $B_x$ can be regarded as public, and so there is conceivable risk that $a_p$ could be back-computed from $b$ if it is fully released. On the other hand, the release of an obfuscated form of $b$ may permit certain research questions to be answered while still ensuring privacy. In order to support the needs of the sustainability research community, it is necessary to understand the relationship between disclosure of $b$ and exposure of elements of $a_p$.

## III. Confidentiality & Privacy Issues

As explained in Section II, $b$ is an emission vector which reports the amount of exchange for each emission during a production or a service. $b$ contains important information both for environmental analysis and marketing decisions. However, LCA practitioners are hesitant to publish $b$ due to their fear of information leakage concerning details of $a_p$, and hence potentially trade secrets that give a specific company a competitive edge over its competition. The question is whether the practitioners are right or not in their concerns. Here, we investigate the possible information leakage out of the publication of $b$. In other words, how much of $a_p$ can be recovered when $b$ is published, given that $B_x$ is public and $b$ is derived from the factorization of $B_x$ and $a_p$ as shown in Equation 2?

### A. Industrial Ecology Privacy Concerns

The operations of an LCA aggregation study is sequential matrix multiplications. If $B_x$ is a nonsingular (invertible) matrix, there exists a unique inverse denoted by $B_x^{-1}$, i.e., $B_x \cdot B_x^{-1} = B_x^{-1} \cdot B_x = I$. Then, Equation 2 has a unique solution, $a_p = B_x^{-1} \cdot b$. This might be seen as a justification of the concern not to publish $b$ along with impact scores, $s$. However, $B_x$ in LCA is a singular matrix most of the time, which means it is not invertible and $a_p$ cannot be solved directly from Equation 2. Is this enough to ensure security guarantees?

The answer to this question is unclear. The concept of Moore-Penrose pseudoinverse of matrices [19], generalizes the notion of a nonsingular (invertible) matrix and makes it applicable to singular matrices. This concept is useful when someone searches for an optimal approximation of a set of linear equation solutions like $A \cdot x = y$, where $A$ is a known $m \times n$ matrix, $y$ is a column vector with $m$ components and $x$ is an unknown column vector. $x$ is the solution for the linear system, which usually leads to the minimum *least square* of $(A \cdot x - y)$. A common approach to compute the pseudoinverse is to use the Singular Values Decomposition (SVD) [11]. This approach can be directly applied in the LCA study to reveal the secret $a_p$ vector with some approximation. The Moore-Penrose pseudoinverse has already been employed to solve different problems like digital imaging methods [4], [24] and astronomical data analysis [20]. A key question is to what extent the $\overline{a}_p$ vector computed from the pseudoinverse allows an attacker to reconstruct $a_p$. The next section investigates the power of the pseudoinverse technique to reveal industry secrets.

### B. Revealing Industry Secrets using Moore-Penrose Pseudoinverse

This section briefly explains the features of the Moore-Penrose pseudoinverse [19] in terms of its capabilities and limitations. The pseudoinverse of a matrix $A$ is denoted by $A^+$. For any matrix $A$, it is known that there exists only one Moore-Penrose inverse $A^+$, i.e., uniqueness. The general psudoinverse solution to a linear system $A \cdot x = y$ is:

$$x = A^+ \cdot y + (I - A^+ \cdot A) \cdot q \tag{4}$$

where $q$ is an arbitrary vector of appropriate order. Since $q$ is arbitrary, there exists an infinite number of solutions when $(I - A^+ \cdot A) \neq 0$. A natural question is whether there is a case where $(I - A^+ \cdot A) = 0$. The answer is in the affirmative when A has a full column rank [18], $A^+ = (A^T \cdot A)^{-1} \cdot A^T$. Having a full column rank guarantees a unique solution to $x$ as seen from the following derivation:

$$\begin{aligned} x &= A^+ \cdot y + (I - A^+ \cdot A) \cdot q \\ &= A^+ \cdot y + (I - (A^T \cdot A)^{-1} \cdot A^T \cdot A) \cdot q \\ &= A^+ \cdot y + (I - I) \cdot q = A^+ \cdot y \end{aligned} \tag{5}$$

In the context of LCA, to the best of our knowledge, having a full column rank in $B_x$ matrix is very rare. The columns are not completely independent from each other which leads to having an infinite number of solutions for the linear system. One can claim that having an infinite number of solutions for $x$ will create enough ambiguity and an adversary will not be able to distinguish which $x$ is close to the original one. However, our empirical studies over a real LCA study disprove this and show that one can solve the linear system approximately close enough using the Moore-Penrose pseudoinverse as we will explain in detail later in Section V. Therefore, we need to ensure the security of publication which prevents an adversary from recovering the solution even with the usage of Moore-Penrose inverse. In the context of privacy-preserving data publication, differential privacy becomes a canonical technique due to its strong privacy guarantees and capability to release useful aggregation information. Given that an LCA study is an aggregation problem, we propose differentially private LCA publications. The next section explains differential privacy and its usage in the context of LCA publication in detail.

## IV. Achieving LCA Privacy

### A. Background: Differential Privacy

Differential privacy provides a strong notion of privacy and is commonly used for statistical data publication [6]. It ensures that the removal or addition of a single record does not significantly affect the outcome of any analysis. It quantitatively bounds how much a single record can contribute to a public output. The formal definition of differential privacy is [6]:

**Definition 1.** A random mechanism M gives $\epsilon$-differential privacy if for any neighboring data sets $D_1$ and $D_2$ differing on at most one element, and all $S \subseteq Range(M)$,

$$Pr[M(D_1) \in S] \leq e^{\epsilon} \cdot Pr[M(D_2) \in S] \tag{6}$$

Differential privacy can be achieved by the addition of random noise. The magnitude of the noise is chosen based on the sensitivity of a query function which considers the largest change in the output of the function with a change of a single record. Such a change is referred to as the *global sensitivity* of a function [6].

**Definition 2.** For any function $f$: $D^n \rightarrow \mathbb{R}^d$, the sensitivity of f is:

$$\Delta f = \max_{D_1, D_2 \in D^n} \parallel f(D_1) - f(D_2) \parallel_1 \tag{7}$$

for all $D_1$, $D_2$ differing in at most one element.

For example, for counting queries, the global sensitivity of a function is 1, since inclusion or exclusion of a single record changes the output of a function by at most 1.

Dwork [6] suggests using the Laplace mechanism to add noise to achieve differential privacy and this has become a canonical approach for differentially private systems. Here, we revisit the differentially private Laplace mechanism.

**Theorem 1.** *The randomized mechanism $M_F$ for a query function $f : D^n \rightarrow \mathbb{R}^d$, computes $f(x)$ and adds a noise sampled from the Laplace distribution to each of the d outputs satisfies $\epsilon$-differential privacy [8]. For such a function, the Laplace mechanism is defined by*

$$M_F(x) = f(x) + (Y_1, Y_2, ..., Y_d) \tag{8}$$

*where $Y_i$ is drawn from the Laplace function $Lap(\Delta f/\epsilon)$.*

A relaxed form of differential privacy, called *approximate differential privacy* or $(\epsilon, \delta)$-*differential privacy* for short, is introduced by Dwork et al. [7]. The approximate differential privacy can be achieved using Gaussian noise calibrated to the $L_2$ sensitivity.

**Definition 3.** $L_2$ sensitivity of a real valued query function $g$: $D^n \rightarrow \mathbb{R}$:

$$\Delta g = \max_{D_1, D_2 \in D^n} \parallel g(D_1) - g(D_2) \parallel_2 \tag{9}$$

for all $D_1$, $D_2$ differing in at most one element.

**Theorem 2.** *The randomized mechanism $M_G$ for a query function g, computes $g(X)$ and adds a noise sampled from the normal distribution $N(\mu,\ \sigma^2)$ where $\mu$ and $\sigma^2$ are mean and variance, respectively. For such a function, the Gaussian mechanism is defined by*

$$M_G(D) = g(D) + N(0, \sigma^2) \tag{10}$$

*where $\sigma = \Delta g \sqrt{2ln(2/\delta)}/\epsilon$. $M_G$ provides $(\epsilon, \delta)$-differential privacy.*

### *B. Differential Privacy for LCA Computation*

The main motivation of this paper is to perform differentially private LCA matrix multiplication in the form of Equation 2, where no adversary is able to recover $a_p$ from the published $b$ vector. Recall that $B_x$ is a publicly known matrix. In this section, we develop two differentially private matrix multiplication mechanisms that will be used later to achieve differentially private publication for LCA computations.

Each element in the $a_p$ vector represents a background process that is included in the production. The privacy goal is to make a publication such that either inclusion or exclusion of a specific background process from the computation has a negligible effect on the output, which is vector $b$. To achieve this goal, differential privacy might be applied by either perturbing the input or the output.

*1) Input Perturbation:* The initial way to achieve differential privacy is to add noise to the input data itself. In the LCA context, the $a_p$ vector contains sensitive information. To achieve $\epsilon$-differentially private computation, the straightforward approach is to generate a differentially private version of $a_p$, and then perform matrix factorization. Similarly, the $(\epsilon, \delta)$-differentially private $a_p$ vector can be published using the Gaussian mechanism, and then it is used in the matrix computation.

In this case, the global sensitivity of the publication considers the maximum change in all possible neighboring vectors.

**Definition 4.** Let $\mathbb{R}$ denote the set of real numbers. For $x_1,\ x_2 \in \mathbb{R}^d$, the sensitivity of the publication:

$$\Delta f_1 = \max \parallel x_1 - x_2 \parallel_1 \tag{11}$$

for all $x_1$, $x_2$ differing in at most one element in the vector.

Assume $x_1^1, x_1^2, .., x_1^d$ are the elements of $x_1$ and $x_2^1, x_2^2, .., x_2^d$ are the elements of $x_2$ such that $\forall i, j \in [1, d]$, $x_1^i, x_2^j \in [0, N]$. If $x_1$ and $x_2$ differ in one element, the maximum change in the publication (global sensitivity) will be $N$.

Although having a data independent sensitivity computation is a desired feature in differentially private publications, the sensitivity computation in our context is data dependent. In theory, the sensitivity is unbounded and can be infinity. Given this fact, differential privacy might be considered as an inappropriate methodology for differentially private LCA computations. However, this is not the case. LCA data modeling has its own characteristics like sparsity, data distribution, which make differential privacy work in the practice for the LCA computations. Later, in this section we will develop a probabilistic estimated variance formulation which is a measurement of utility of an LCA publication.

Now, we can formally define our differentially private vector publication mechanism.

**Proposition 1.** *The randomized mechanism $M_K$ that outputs the following vector is $\epsilon$-differentially private:*

$$M_K(x) = x + k \tag{12}$$

*where k is a vector consisting of $n$ independent samples drawn from the Laplace distribution function with a scale $\Delta f_1/\epsilon$, i. e., $Lap(\Delta f_1/\epsilon)$.*

*Proof.* Recall that $x$ is a vector consisting of the true answers. $M_K$ mechanism adds independent Laplace noise to each element of $x$. Thus, the output of $M_K$ is a vector of length $d$ containing a noisy answer for each element in $x$. The $M_K$ mechanism incorporates the features of Theorem 1, hence, satisfies $\epsilon$-differential privacy. □

Recall that, our motivation is to publish vector $b$ in LCA computation, not $a_p$. Using the $M_K$ mechanism, it is possible to publish $\epsilon$-differentially private $a_p$. Now, the differentially private version of $a_p$ will be used to compute resulting $b$ vector.

**Proposition 2.** *Given a public $A \in \mathbb{R}^{m \times n}$ and private $x \in \mathbb{R}^n$, the randomized mechanism $M_{F_1}$ that performs the following operation ensures $\epsilon$-differentially privacy for x:*

$$M_{F_1}(A, x) = A \cdot M_K(x) \tag{13}$$

*Proof.* $M_{F_1}$ uses a differentially private (obfuscated) version of $x$, generated by mechanism $M_K$, in the matrix factorization.

$A$ is known by the public and transforming to the LCA computation $A$ stands for a $B_x$ matrix where the rows are emissions and the columns are processes that are included in the study. The mechanism has to ensure that either the inclusion or removal of a process should not reveal any information about the process. It is already proven that the $M_K$ mechanism ensures differential privacy. The factorization of $A.M_K(x)$ is a post processing over the differentially private $x$. The factorization does not have any access to the original $x$ matrix, hence it does not violate differential privacy. Although the mechanism outputs only the result of the factorization, assume that an adversary tries to find the original $x$ vector by solving the linear system. In the best case, the adversary will get $M_K(x)$ by solving the linear system which is already proven $\epsilon$-differentially private. Therefore, $M_{F_1}$ mechanism ensures $\epsilon$-differential privacy for $x$. □

**Expected Variance of Error.** To measure the utility, we analyze the accuracy of resulting vector. Let $y$ denote the factorization of $A.M_K(x)$ where $y_1, y_2, .., y_m$ are the elements of $y$. We use $y_i$ to denote the correct value, $\widehat{y_i}$ to denote differentially private result, and $E_{M_{F_1}}(y_i)$ to denote the absolute error for $y_i$ with $M_{F_1}$ mechanism such that:

$$E_{M_{F_1}}(y_i) = |y_i - \widehat{y_i}| \tag{14}$$

Given each $y_i$ is randomized, $E_{M_{F_1}}(y_i)$ is a random variable. Since $y$ has $m$ elements, the average variance of error (the mean squared error) of $M_{F_1}$ is:

$$\mathrm{Var_{avg}}(M_{F_1}) = \frac{\sum_{k=1}^{m}(E_{M_{F_1}}(y_k))^2}{m} \tag{15}$$

In the $M_{F_1}$ mechanism, each element of x is added a noise sampled from the Laplace distribution $Lap(\Delta f_1/\epsilon)$. The variance at each element, therefore, $\mathrm{Var_e} = 2.(\frac{\Delta f_1}{\epsilon})^2$. Note that the sampled random variables are uncorrelated. In the factorization, for each row, the $j^{th}$ element of $A$ is multiplied by the $j^{th}$ element of obfuscated $x$.

$$\mathrm{Var_i(E_{M_{F_1}}(y_i))} = \sum_{k=1}^{n} A_{ik}^2.\mathrm{Var_e} \tag{16}$$

In the factorization, each element of $A$ is a weighting constant. It corresponds to the $B_x$ matrix in LCA computations. Without modeling the LCA study completely, it is possible to estimate $A_{ik}$ if the underlying data distribution of $A$ is known. $A$ consists of $m \times n$ discrete values, we can define the probability density function $g(z_i)$, such that for any $z_i$, which is a value that $Z$ can take, $g$ gives the probability that the random variable $Z$ equals $z_i$:

$$\begin{gathered} P(Z = z_i) = g(z_i) \quad i = 1, 2, ... \\ g(z_i) \geq 0, \sum_i g(z_i) = 1 \end{gathered} \tag{17}$$

Then, the expected value for $Z$ is:

$$\mathrm{Ep}(Z) = \sum_z z.g(z) \tag{18}$$

Using the expected value for any entry in $A$, we can compute the expected error variance of $y$'s elements in the following way.

$$\mathrm{Ep(Var_i(E_{M_{F_1}}(y_i)))} = n.\mathrm{Ep}(Z)^2.\mathrm{Var_e} \tag{19}$$

The final step is to compute the expected average error variance for the $M_{F_1}$ mechanism.

$$\mathrm{Ep_{avg}}(\mathrm{Var_{avg}}(M_{F_1})) = \frac{n^2.\mathrm{Ep}(Z)^4.\mathrm{Var_e}}{m} \tag{20}$$

The expected average error variance depends on the data distribution of $A$, and there is no boundary for the error. However, in the LCA context, the $B_x$ matrix is sparse most of the time and most of the entries are either zero (0) or close to zero, which makes the expected average error variance low. Although unbounded sensitivity is a problem, the characteristics of LCA publications enable differentially private publication to deliver significant utility, which will later be discussed and verified in Section V-B.

*2) Output Perturbation:* To achieve differential privacy by perturbing the output, the desired differentially private mechanism initially computes the function, and then adds noise to each element of the computed output to obtain differentially private publication. Similar to the previous setting, $A$ is a public matrix and $x$ is a private vector which we want to preserve its privacy.

**Definition 5.** Let $\mathbb{R}$ denote the set of real numbers where $A \in \mathbb{R}^{m \times n}$ and $x \in \mathbb{R}^n$. A matrix multiplication function $f$:$\mathbb{R}^{m \times n} \times \mathbb{R}^n \rightarrow \mathbb{R}^m$ is defined by:

$$f(A, x) = A \cdot x \tag{21}$$

The output of $f$ is an $m$-dimensional vector. To achieve differentially private matrix multiplication, the noise should be generated based on the sensitivity of $f$. The sensitivity of $f$ considers the maximum change in the output with a single change in the vector $x$. The defined function is a matrix multiplication, thus, a single change in $x$ will result in changes in every entry of the output. We consider the maximum change as a sensitivity with a single change.

**Definition 6.** For $x_1, x_2 \in \mathbb{R}^n$, $A_1, A_2 \in \mathbb{R}^{m \times n}$, the sensitivity of $f(A, x)$:

$$\Delta f_2 = \max \parallel f(A_1, x_1) - f(A_2, x_2) \parallel_1 \tag{22}$$

for all $x_1$, $x_2$ differing in at most one element.

When an element is excluded from the $x$ vector, the corresponding column is also excluded from $A$ to perform matrix multiplication. For example, if the second entry from the $x$ vector is excluded, the second column of matrix $A$ should also be removed from $A$ to perform multiplication operation consistently. Basically, this is an exclusion of a process from an LCA model and observation of its effect.

In the proposition below, we define a differentially private matrix multiplication mechanism.

**Proposition 3.** *Given a matrix multiplication function $f(A, x)$, the randomized mechanism $M_{F_2}$ that outputs the following vector is $\epsilon$-differentially private:*

$$M_{F_2}(A, x) = f(A, x) + k \tag{23}$$

*where k is a vector consisting of $m$ independent samples drawn from the Laplace distribution function with a scale $\Delta f_2/\epsilon$, i. e., $Lap(\Delta f_2/\epsilon)$.*

$M_{F_2}$ initially executes $f(A, x)$ which outputs the multiplication of $A$ with $x$. Then, the mechanism adds a randomly sampled vector $k$ to the result of the multiplication to obfuscate it.

*Proof.* $M_{F_2}$ incorporates the features of Theorem 1, which states that a random mechanism satisfies $\epsilon$-differential privacy *iff* each output of a function is added a noise sampled from the Laplace distribution. $M_{F_2}$ initially, performs matrix multiplication, and then adds a noise to each element in the resulting vector. Therefore, $M_{F_2}$ is $\epsilon$-differentially private. □

**Expected Variance of Error.** Let $y$ denote the result of $M_{F_2}(A, x)$. The absolute error is caused only by the addition of random noises sampled from the Laplace distribution. Therefore, the error variance of $y$'s entries:

$$\mathrm{Var_i}(\mathrm{E_{M_{F_2}}}(\mathrm{y_i})) = \mathrm{Var_e} \tag{24}$$

where $\mathrm{Var_e} = 2.(\frac{\Delta f_2}{\epsilon})^2$ for $M_{F_2}$.

Since all noises are independently generated and have the same variance, the average error variance is:

$$\mathrm{Var_{avg}}(\mathrm{E_{M_{F_2}}}(\mathrm{y_i})) = \frac{\mathrm{Var_e}}{m} \tag{25}$$

The average error variance is again data dependent, but as it will be verified with a real life example in the next section, it is likely for LCA computations to preserve the utility of differential privacy.

## V. Evaluation of Privacy-Preserving LCA computation

To evaluate the security concerns and challenges of an LCA publication and the effects of differential privacy, we conducted experiments over a real LCA study for *distillers grain*. Using U.S. Life Cycle Inventory (USLCI) [1], we design and build a case study for distillers grain.

**Data sets:** The distillers grain study contains 39 background processes and 378 elementary flows. Therefore, $a_p$ is a 39-dimensional vector and $B_x$ is a 378×39 matrix. The distinctive property of this data set is having a very broad range of numbers. The entries in the matrices range from $10^{-15}$ to $10^3$. We will later explain the effects of having numbers from such a wide range.

This section initially presents attacks to demonstrate whether there is a need for privacy preserving publication in reality, given that the only motivation is to make $b$ public. Due to the special properties of the LCA data, the answer to this inquiry is affirmative. Therefore, the section continues with the detailed guideline on applying differential privacy to an LCA computation where the aim is to make the publication useful (high utility) while still preserving the privacy.

### A. Attack against LCA publication with Public $b$

The attack is formed to understand the security and privacy breaches in LCA publication. Suppose an LCA practitioner wants to publish $b$. She computes $b$ using Equation 2 and makes it publicly available while not providing any information about $a_p$. As stated before, $B_x$ is publicly known. The LCA practitioner thinks the computation is secure, since $B_x$ is a singular matrix and there is no way to recover $a_p$. An adversary, on the other hand, is interested in learning information about $a_p$, since this vector contains confidential information regarding production processes which could be used to gain financial benefits.

*a) Attack with Pseudoinverse::* The attacker develops its attack by computing the Moore-Penrose pseudoinverse of $B_x$ which is covered in Section III-B. The rank of $B_x$ is 29 -not a full column rank-. This means the solution to the $B_x \cdot a_p = b$ linear system is not unique. The common approach to resolve this issue uses the least square approach to optimize the approximation for $a_p$. This will output an approximate solution that is denoted by $\widehat{a_p}$. There are variety of ways to measure the distance between two vectors all of which might provide different results. To measure the closeness of the output, the Euclidean distance is used in this study. Additionally, the computations provide details about how many entries in the vectors are close within a given threshold. We use *close enough* as a term to express that the distance between the approximated value and the actual value is less than a provided threshold, which basically means an adversary approximate enough to recover the actual value. For example, consider a scenario where the first entry in $a_p$ is 3 and the attacker finds the first entry of $\widehat{a_p}$ to be 2. If the threshold is 0.5, the comparison indicates that the outputs are far away from each other and this is a failure for the attacker. However, if the threshold is 2, the attacker recovers the entry approximate enough and this is a success.

When the attack is executed[1] using the distillers LCA data, the distance between the actual $a_p$ and the computed vector $\widehat{a_p}$, i. e., $\parallel a_p - \widehat{a_p} \parallel_2$, is 0.6558. Although the distance seems close, the attacker is able to approximate only 3 processes out of 39 close enough when the threshold is $10^{-10}$. However, this is still a good source of information to claim that a practitioner is not able to secure $a_p$ completely and the data is breached.

Furthermore, in a piratical setting, many entries of the $a_p$ vector are 0 for the distillers grain data set. It is reasonable to assume that an expert in the field has enough background knowledge to estimate which processes are included in the

[1] The Singular Value Decomposition technique is used to compute the pseudoinverse.

computation pretty well. In such a case, the expert can develop a stronger attack against the publication of $b$ and hence knowledge of $a_p$ by removing all zero entries from $ap$ which will result in the removal of the corresponding columns in $B_x$. In our study, 19 entries of $ap$ are 0. When these entries are removed from the computation, the attacker has 20 entries to estimate. The new $B_x$ matrix does not have a full column rank (it is 17). When the attacker solves the linear system using the pseudoinverse technique, the distance $\parallel a_p - \widehat{a_p} \parallel_2$ equals $0.15559$. Compared to the initial case, it is a more powerful attack and the attacker is able to approximate 13 processes out of 20 when the threshold is $10^{-10}$. Given that the attacker already knows the zero entries, she manages to recover almost $82.05\%$ of $a_p$. The conducted experiments outline the power of pseudoinverse approach in the context of LCA domain. The important reasoning for such a good approximation is the domain range of the LCA data. The case study contains many very small numbers and this helps in approaximating the result better.

These attacks show that publishing $b$ without securing with any privacy technique has severe security issues and the concerns over making $b$ public in the LCA community are justified. Therefore, the publication should be made privacy-preserving. Next, this work applies different differential privacy techniques to secure the publication. The publications are again attacked by the adversaries to test the security of the publication in practice.

### B. *Differentially Private LCA Computation*

In this section, we explain how to perform differentially private LCA computation efficiently by using the mechanisms that are introduced in Section IV-B and evaluate the efficiency of the publication in terms of utility and security. The main metric to provide comparison is again the *Euclidean distance* of matrices for both utility and security. To ensure better utility, it is better to have a smaller distance between the original and the computed matrices. However, it is desired to have a larger distance between matrices to achieve better security.

Given $B_x$ and $a_p$, the randomized mechanism $M_{F_1}(B_x, a_p)$ ensures $\epsilon$-differentially private matrix multiplication by perturbing $a_p$ first and then factorizing it with $B_x$ (Proposition 2). $\widehat{b}$ denotes the obfuscated version of $b$ vector. The LCA practitioner publishes the obfuscated version and keeps any version of $a_p$ private. If an adversary solves the linear system of $B_x \cdot \widehat{a_p} = \widehat{b}$ perfectly, she ends up having $\widehat{a_p}$ int the best case. Since $\widehat{a_p}$ is $\epsilon$-differentially private, privacy is still guaranteed.

Table I presents the results of privacy-preserving LCA computation with the $M_{F_1}$ mechanism. The experiments are conducted by varying the $\epsilon$ security parameter. $\Delta b$ denotes the Euclidean distance between the original $b$ vector and $\widehat{b}$. $\Delta\widehat{a_p}$ measures the distance between $a_p$ and $\widehat{a_p}$ where $\widehat{a_p}$ is the output of the $M_K$ mechanism (Proposition 1). This explicitly depicts the effect of random noise addition. Assume that an adversary finds an approximate solution, denoted by $\overline{a}_p$, to $B_x \cdot \widehat{a_p} = \widehat{b}$ using the pseudoinverse approach. $\Delta a_p$ is defined as the Euclidean distance between $a_p$ and $\overline{a}_p$.

Table I
$M_{F_1}$ Mechanism For Matrix Factorization

| $\epsilon$ | $\Delta b$ | $\Delta\widehat{a_p}$ | $\Delta a_p$ |
|---|---|---|---|
| 0.01 | 180.1E3 | 498.934 | 447.95 |
| 0.05 | 94216.22 | 94.86 | 91.449 |
| 0.1 | 128531.08 | 32.099 | 29.876 |
| 0.5 | 14580.31 | 5.348 | 5.144 |
| 1 | 5393.87 | 7.66 | 7.247 |
| 2 | 3840.31 | 2.03 | 1.783 |
| 10 | 2727.62 | 0.464 | 0.44 |
| 100 | 247.28 | 0.044 | 0.162 |

Table II
$M_{F_2}$ Mechanism For Matrix Factorization

| $\epsilon$ | $\Delta b$ | $\Delta a_p$ |
|---|---|---|
| 0.01 | 3311.5 | 9.91E8 |
| 0.05 | 697.663 | 8.23E7 |
| 0.1 | 309.63 | 8.98E7 |
| 0.5 | 69.421 | 1.11E7 |
| 1 | 40.601 | 222.4E4 |
| 2 | 17.467 | 509.8E4 |
| 10 | 2.865 | 634.2E3 |
| 100 | 0.339 | 102.1E3 |

It is a well-known fact that when $\epsilon$ is small, the amount of noise addition is larger but ensures more security [17]. As $\epsilon$ increases, less noise is added which results in more utility. Finding the correct $\epsilon$ value for differentially private systems is a well studied research problem [17], [13]. $\ln 2$ and $\ln 3$ are widely used $\epsilon$ values for differentially private applications. This suggestions are also applicable in our context. We assume $\epsilon \approx 1$ is an ideal setting in our context.

When $\epsilon$ is $0.01$, the distance between differentially private $a_p$ and the original $a_p$ is maximum, $498.934$. When $\epsilon$ is 1, this distance is $7.66$, which is also not very small. It is easy to infer that the noises are sampled with a large scale from the Laplace distribution. The main reason for this is that the values of $a_p$ range from $6.48 \times 10^{-8}$ to $0.7$. In order to hide the existence of a single record, the differential privacy mechanism adds large noises since the sensitivity is too high.

The change in $b$ is relatively large as a result of the $M_{F_1}$ mechanism. It seems that the small perturbations in $a_p$ introduce large perturbations in $b$. Such a system is referred to as *ill-conditioned* [3]. When $\epsilon$ is 1, $\Delta b$ is 5393.87. This can be inferred as too much utility loss. However, when the result of the computation is analyzed in detail, 165 elements out of 378 (44%) are approximately close within the threshold of $10^{-10}$ when $\epsilon$ is 1. If an analyst wants to make a study for individual emissions, such a publication is very useful. The other important feature of this publication is its privacy. When an attacker executes the attack described before, she cannot recover $a_p$ at all. The attacker computes $\overline{a}_p$ which has a distance of 7.247 from $a_p$. More importantly, even if she knows the location of all zero elements in the vector, she cannot approximate even 1 element out of 20 within a threshold of $10^{-10}$. This validates the strong privacy guarantee of the $M_{F_1}$ mechanism.

Table III
$M_{F_1}$ Mechanism For Matrix Factorization with Normalization

| $\epsilon$ | $\Delta b$ | $\Delta \widehat{a_p}$ | $\Delta a_p$ |
|---|---|---|---|
| 0.01 | 122057.076 | 337.99 | 303.475 |
| 0.05 | 63824.391 | 64.355 | 62.082 |
| 0.1 | 87070.216 | 21.647 | 20.13 |
| 0.5 | 9877.376 | 5.176 | 4.901 |
| 1 | 3653.79 | 3.677 | 3.538 |
| 2 | 2601.958 | 1.427 | 1.271 |
| 10 | 1848.137 | 0.367 | 0.375 |
| 100 | 167.92 | 0.259 | 0.299 |

Table IV
$M_{F_2}$ Mechanism For Matrix Factorization with Normalization

| $\epsilon$ | $\Delta b$ | $\Delta a_p$ |
|---|---|---|
| 0.01 | 1609.384 | 4.8E8 |
| 0.05 | 339.06 | 4.0E7 |
| 0.1 | 150.48 | 4.36E7 |
| 0.5 | 33.738 | 5.41E6 |
| 1 | 19.73 | 1.08E6 |
| 2 | 8.489 | 2.47E6 |
| 10 | 1.392 | 3.08E5 |
| 100 | 0.164 | 4.9E4 |

To achieve a differentially private LCA publication with the output perturbation, the $M_{F_2}$ mechanism is proposed (Proposition 3). This approach initially computes $b$ by multiplying $B_x$ with $a_p$, and then obfuscates $b$ by adding a random noise vector. $a_p$ is again kept secret and the obfuscated emission vector $\widehat{b}$ is made public.

Table II presents the experimental results of privacy-preserving LCA computation with the $M_{F_2}$ mechanism. This kind of publication reduces utility less compared to the earlier publication with $M_{F_1}$ when $\Delta b$ results are considered. When $\epsilon$ is 1, $\Delta b$ equals $40.601$ when $M_{F_2}$ is used. It is $5393.87$ when the publication is done with the $M_{F_1}$ mechanism for the same $\epsilon$. However, when the results are analyzed in detail, none of the entries in $\widehat{b}$ is close enough to the entries in $b$ within the threshold of $10^{-10}$. As explained before, in a similar setting, the $M_{F_1}$ outputs $44\%$ of the entries close enough. The trade-off between $M_{F_1}$ and $M_{F_2}$ can easily be seen by considering the empirical studies. The $M_{F_1}$ delivers better utility for an analysis of individual emissions. On the other hand, $M_{F_2}$ delivers better utility if an analysis contains aggregate computation, e. g., "What is the summation of emissions $(b_i, b_j, b_k, ..., b_n)$ in the distillers grain study?".

In terms of privacy, $M_{F_2}$ achieves a strong privacy as presented in Table II. When $\epsilon$ is 1, $\Delta a_p$ equals $222.4 \times 10^4$. When an attacker tries to solve the system with the pseudoinverse approach, the computed $\overline{a}_p$ has a distance of $222.4 \times 10^4$ to the original $a_p$. The attacker is not able to approximate any entries in $a_p$.

Both $M_{F_1}$ and $M_{F_2}$ ensures strong privacy. This is very positive and convincing findings in the context of LCA publication. The practitioners can feel confident about publishing $b$. Although the current techniques deliver reasonable utility, the question remains whether there is a way to improve utility without sacrificing the privacy guarantees in such *ill-conditioned* systems.

To answer this inquiry, this study explores a normalization technique to decrease the utility loss in the LCA computation. The motivation for applying normalization is narrowing down the range of numbers that data sets have, since such a wide range causes differentially private systems to inject more noise to the system.

Table III presents the results when $M_{F_1}$ is executed with the normalized version of $a_p$, denoted by $\tilde{a}_p$. In this computation, $\tilde{a}_p$ is provided as an input instead of $a_p$. As seen from the results, using $\tilde{a}_p$ instead of $a_p$ decreases the distance between the published $\widehat{b}$ and $b$ from 5393.87 to 3653.79, since $\widehat{a_p}$ contains less noise compared to the non-normalized computation. In addition to that 167 entries of $\widehat{b}$ are approximate enough to the entries of $b$ within a threshold of $10^{-10}$. It is 165 if the non-normalized input is used in the computation. Therefore, it is reasonable to state that using the normalized input increases the utility of the $M_{F_1}$ mechanism.

To apply a similar approach to $M_{F_2}$, the normalization operation is performed on $B_x$, where the normalized version is denoted by $\tilde{B}_x$. $\tilde{B}_x$ and $a_p$ are inputs to the $M_{F_2}$ mechanism. The results of the publications are presented in Table IV.

The normalization approach also has a positive effect on the $M_{F_2}$ mechanism in terms of utility. Since the system is *ill-conditioned*, narrowing down the range of numbers results in adding less noise to the output of the publication. When $\epsilon$ equals 1, $\Delta b$ is 19.73, in contrast to 40.601 when normalization is not employed. This is a huge gain in the utility. However, the normalization technique does not improve the utility of the $M_{F_2}$ mechanism in terms of individual emission analysis. None of the emissions is close enough within the threshold of $10^{-10}$ to the original emissions.

The normalization does not have any negative impact on the privacy for both $M_{F_1}$ and $M_{F_2}$. An adversary cannot approximate any element in $ap$.

Considering the overall empirical study, differentially private LCA computation can be achieved with either $M_{F_1}$ or $M_{F_2}$ without sacrificing security. Although $M_{F_1}$ is useful for an individual emission analysis, $M_{F_2}$ delivers good utility for aggregate analysis on $b$. The straightforward application of normalization increases the utility.

## VI. Conclusion

In this paper, we present a comprehensive study to explore the privacy concerns over publicizing the industrial activities in the form of LCA computations. Accurate and high quality sustainability assessment requires detailed information about industrial activities; however such information is considered confidential. This paper initiates a study to explore privacy and security challenges that prevent organizations from making public disclosures about their activities. Our empirical studies show that the application of privacy-preserving techniques is required to preserve the privacy of private data. Otherwise, it is possible to expose the private data by reverse-computing from the publication. To support the needs of the sustainability research community, this paper proposes differentially private LCA computations and explains how to achieve it for LCA computations by either perturbing the input data or the output data. Our evaluations on a real LCA example from a distillers grain study demonstrates that the use of differential privacy to publish more detailed information ensures strong privacy while revealing useful information for analysts.

## ACKNOWLEDGMENT

This work is funded by an NSF grant CCF-1442966.

## REFERENCES

[1] U.S. Life Cycle Inventory Database, 2012. National Renewable Energy Laboratory, 2012. Accessed March 11, 2016: https://www.lcacommons.gov/nrel/search.

[2] M. Baitz, C. M. Colodel, T. Kupfer, J. Florin, O. Schuller, F. Hassel, M. Kokborg, A. Köhle, D. Thylmann, A. Stoffregen, S. Schöll, J. Görke, and M. Rudolf. Gabi database & modelling principles 2013. Technical report, PE International AG, 2013.

[3] D. A. Belsey, E. Kuh, and R. E. Welsch. *"The condition Number". Regression diagnostics: Identifying influential data and sources of collinearity*. John Wiley, 1980.

[4] S. Chountasis, V. N. Katsikis, and D. Pappas. Applications of the moore-penrose inverse in digital image restoration. *Mathematical Problems in Engineering*, 2009:Article ID 170724, 12 p.–Article ID 170724, 12 p., 2009.

[5] M. A. Curran. *Environmental Life-Cycle Assessment*. McGraw-Hill Professional Publishing, July 1996.

[6] C. Dwork. *Automata, Languages and Programming: 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II*, chapter Differential Privacy, pages 1–12. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[7] C. Dwork, K. Kenthapadi, F. McSherry, and I. Mironov. Our data, ourselves: Privacy via distributed noise generation. In *Advances in Cryptology (EUROCRYPT 2006)*, volume 4004, pages 486–503, Saint Petersburg, Russia, May 2006. Springer Verlag.

[8] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the Third Conference on Theory of Cryptography*, TCC'06, pages 265–284, Berlin, Heidelberg, 2006. Springer-Verlag.

[9] G. Finnveden, M. Z. Hauschild, T. Ekvall, J. Guinée, R. Heijungs, S. Hellweg, A. Koehler, D. Pennington, and S. Suh. Recent developments in life cycle assessment. *Journal of Environmental Management*, 91(1):1–21, 2009.

[10] R. Frischknecht. Transparency in LCA-a heretical request? *Int J LCA*, 9(4):211–213, jul 2004.

[11] G. H. Golub and C. Reinsch. Singular value decomposition and least squares solutions. *Numerische mathematik*, 14(5):403–420, 1970.

[12] R. Heijungs and S. Suh. *The computational structure of life cycle assessment*, volume 11. Springer Science & Business Media, 2002.

[13] J. Hsu, M. Gaboardi, A. Haeberlen, S. Khanna, A. Narayan, B. C. Pierce, and A. Roth. Differential Privacy: An Economic Method for Choosing Epsilon. *arXiv:1402.3329 [cs]*, Feb. 2014. arXiv: 1402.3329.

[14] R. G. Hunt and W. E. Franklin. LCA — how it came about. *International Journal of Life Cycle Assessment*, 1(1):4–7, 1996.

[15] ISO 14044. *Environmental management — Life cycle assessment — Requirements and guidelines*. ISO, Geneva, Switzerland, 2006.

[16] B. Kuczenski. Partial ordering of life cycle inventory databases. *The International Journal of Life Cycle Assessment*, 20(12):1673–1683, Oct 2015.

[17] J. Lee and C. Clifton. How much is enough? choosing $\epsilon$ for differential privacy. In *Information Security, 14th International Conference, ISC 2011, Xi'an, China, October 26-29, 2011. Proceedings*, pages 325–340, 2011.

[18] J. R. Magnus, H. Neudecker, et al. Matrix differential calculus with applications in statistics and econometrics. chapter Kronecker products, the vec operator and the Moore-Penrose inverse. John Wiley & Sons, 1995.

[19] E. H. Moore. On the reciprocal of the general algebraic matrix. *Bulletin of the American Mathematical Society*, 26:394–395.

[20] J. Steiner, R. Menezes, T. Ricci, and A. Oliveira. Pca tomography: how to extract information from data cubes. *Monthly Notices of the Royal Astronomical Society*, 395(1):64–75, 2009.

[21] A.-M. Tillman. Significance of decision-making for LCA methodology. *Environ. Impact Assess. Rev.*, 20(1):113 – 123, 2000.

[22] UNEP/SETAC. Global guidance principles for life cycle assessment databases. Technical report, United Nations Environment Programme, 2011.

[23] United Nations. The millennium development goals report 2015. http://www.un.org/millenniumgoals/, 2015.

[24] R. Van de Walle, H. H. Barrett, K. J. Myers, M. Aitbach, B. Desplanques, A. F. Gmitro, J. Cornelis, and I. Lemahieu. Reconstruction of mr images from data acquired on a general nonregular grid by pseudoinverse calculation. *Medical Imaging, IEEE Transactions on*, 19(12):1160–1167, 2000.

[25] B. P. Weidema, C. Bauer, R. Hischier, C. Mutel, T. Nemecek, J. Reinhard, C. Vadenbo, and G. Wernet. Overview and methodology. data quality guideline for the ecoinvent database version 3. Technical report, The ecoinvent Centre, St. Gallen, 2013.

[26] T. O. Wiedmann, H. Schandl, M. Lenzen, D. Moran, S. Suh, J. West, and K. Kanemoto. The material footprint of nations. *Proceedings of the National Academy of Sciences*, 2013. Early publication; doi:10.1073/pnas.1220362110.