Privacy-preserving tax calculations in smart cities by means of Inner-Product Functional Encryption

Oana Stan, Renaud Sirdey, Cedric Gouy-Pailler, Pierre Blanchart, Amira Benhamida, Mohamed-Haykel Zayani

Privacy-preserving tax calculations in smart cities by means of Inner-Product Functional Encryption

Oana Stan*†, Renaud Sirdey*†, Cédric Gouy-Pailler*†,
Pierre Blanchart*†, Amira Ben Hamida†, Mohamed-Haykel Zayani†
*CEA, LIST,
91191 Gif-sur-Yvette Cedex, France
Email: name.surname@cea.fr
†IRT SystemX
8, av. de la Vauve, 91120 Palaiseau, France
Email: name.surname@irt-systemx.fr

***Abstract*—Functional encryption is a recent generalization of public-key cryptography which aims at enabling secret-key owners to decrypt only functions of the encrypted data. This model is very promising in terms of applications. Yet, although general constructions of theoretical interests do exist, practical functional encryption is presently limited to the evaluation of low-degree functions of the encrypted inputs. In this paper, we investigate how Inner-Product Functional Encryption (IPFE) may enable the design of tax calculation system with built-in privacy. The paper is also concluded by performances results demonstrating the practicality of the approach on the concrete issue of carbon tax calculations.**

1. Introduction

Smart cities raise several technological challenges and innovations, in the field of smart manufacturing but also over transport, energy and resources. One of the key components of the future smart cities is their energy efficiency while providing innovative solutions to the climate change. Environmental sustainability, through an appropriate waste and energy management, is one of the domain for which smart cities will have to develop new efficient services.

In this context, we investigate a new kind of digital environmental taxation service, addressed to smart factories, which can be deployed into a smart city and accessed remotely, while assuring data privacy. Due to its promise of a connected environment for all the citizens and businesses, smart cities are exposed to a diverse set of data privacy breaches, criminal misuses and security threats. That is the reason we consider it is of primary importance to take into account data confidentiality from the beginning and develop a privacy-by-design service. The cryptographic technique underlying our approach is functional encryption. Informally, using this type of public-key encryption schemes, the decryption key allows a user to learn a function of the encrypted data without learning anything else about the data. As such, it is thus possible to compute on encrypted data and to recover the result in clear form.

In this paper, we focus on the application of the special case of Inner-Product Functional Encryption [1] (IPFE). This type of functional encryption, relatively simple yet surprisingly powerful, allows only restricted computations but is more efficient than the general constructions proposing functional encryption for any computation which remains mostly of theoretical interest.

The main contributions of this work are the following: i) a new private-by-design service of taxation for smart factories, based on functional encryption; ii) the application of the general approach for the use case of carbon tax, using an Inner-Product Functional Encryption; iii) instantiations and experimental results with the IPFE scheme from [1] under Decision Diffie-Hellman (DDH) and, respectively, Decision Composite Residuosity (DCR) assumptions.

2. Use case description

Environmental sustainability is a key challenge for modern cities. It relies on the responsability of various actors, ranging from citizens to private companies. Yet this empowerment process can also greatly benefit from behavior-based taxation models. This taxation approach consists in financially penalizing individual behaviors depending on their environmental impact. A substantial barrier to this approach is that it entails quantitatively measuring those behaviors. While technically feasible, it poses serious issues in terms of privacy. We advocate that functional encryption could serve as a key enabler for such a model. Specifically, the addressed use case concerns a behavior-based taxation model for smart factories accessible remotely and hosted by a recognized tax entity. For privacy reasons, the data produced by the smart factory which serves in the computation of the tax revenue, is encrypted (with functional encryption). The party responsible with the taxes collects the encrypted smart factory data, computes the taxes amount using a pre-established model and takes the necessary actions (i.e. ask for the payment). In this framework, we consider that a third party authority is in charge of designing and setting up the taxation model. Also, consistently with the functional

encryption setting, it is the one which performs the keys generation and management.

Here, we present a taxation model for environmental pollution and, more precisely, on carbon dioxide. It is relatively easy to apply this taxation model for the production of a smart factory since the total amount of carbon emissions depends linearly on the emissions made in the different sources of energy used in the production process. As such, one can use IPFE to compute the carbon tax for a given smart factory, without having access to its data related to the emissions and the energy sources it uses.

3. State of art

Functional encryption is a recent more flexible cryptographic framework allowing to go beyond the classical encryption schemes proposing an "all-or-nothing" decryption. Within FE schemes, a master authority can distribute keys which gives the possibility to perform some computations over encrypted data in a controlled way. Given a ciphertext c with underlying plaintext x and a secret key sk_f associated with a function f, it is possible to recover, when decrypting c, only the result of the evaluation of $f(x)$ without learning anything else about x.

Functional encryption is an alternative to other proposals for secure cloud computing, such as secure Multi-Party Computation (MPC) [2], [3] or Fully Homomorphic Encryption (FHE) [4], [5]. Moreover, FE generalizes several types of encryption such as identity-based encryption [6], [7], fuzzy identity-based encryption [8], attribute-based encryption [9], [10], predicate encryption [11], [12], broadcast encryption [13], etc.

In recent years, the focus has been in designing efficient schemes for restricted classes of functions or polynomials, such as linear [1], [14] or quadratic [15] ones. In the sequel, we concentrate on the inner-product functional encryption which is one of the basic yet powerful examples of practical FE constructions, build under well-understood security assumptions.

3.1. Inner Product Functional Encryption

In an IPFE scheme the messages are expressed as vectors such that, given the encryption of a vector x, and a generated key associated with a vector y, one can obtain upon decryption only the inner product $< x, y >$. Even if it seems limited in expressivity, IPFE can find many natural applications, as we will show later on in this paper.

The (inner product) Functional Encryption protocol involves three parties:

- A key owner, say Authority, who generates mpk, msk and sk_x.
- An owner of a vector x, say Operator, who provide that vector to the key owner, receive sk_x from the latter and evaluates the inner product on the ciphertexts it receives from the following parties.
- Several owners of vector y's, say User's, who retrieve mpk from the key owner, use it to encrypt their data and send them to the previous party (who owns the vector x) for evaluation.

Then Operator can evaluate the dot-products of vector x with any number of encrypted vector y's sent to him by the User's and has *by construction* access only to the result of that dot product evaluation. On the other hand, Authority has no access to the y vectors. Of course, and this is intrinsic to the FE security model, Operator and Authority must not collude.

As a more concrete example, we could imagine that Operator is a pharmaceutical company, that Authority is a national health authority (e.g. the FDA in the US or the ANSM in France) and that the User's are patients. Then, the pharmaceutical company would be able to run a epidemiological study, approved by the relevant authority, without having access to the raw patient data.

Still, as already emphasized, practical FE schemes are presently limited to very restricted classes of computations: mostly inner-product and degree-2 polynomials. However, as we shall see in the sequel this is already enough to addresse a number of practically relevant use-cases.

3.2. FE vs FHE

An additional point, which is worth emphasizing, is the difference between Fully Homomorphic (FHE) and Functional Encryption (FE). In FE, as was just discussed, the server is able to compute a function f over encrypted data and has access to the final result of the evaluation of that function (and only to that result). This means that the decryption capability has to be intricated to f and that the server must reveal the inner working of its algorithm to the user (or to a trusted third party) during the setup of the system. In FHE, however, the server can compute any algorithms (including algorithms not known at the time of encryption) but as a consequence of the all-or-nothing decryption property of FHE cannot have access whatsoever to any results. To make this a little bit more concrete, using FE, one could in principle implement e.g. selective packet routing with hidden criteria (the criterion would have to be known to a trusted authority but not to the end user) whereas in FHE one cannot do so because the evaluation of the routing criteria would remain sealed in the cryptosystem and decryption, in the FHE model, is all-or-nothing. So, these are two different settings but understanding both allows to better grasp their applicability. Also, FHE is quite often used as a building-block of general (yet not practical as of today) FE schemes.

3.3. Pollution and environmental taxation models

Between the main advantages of environmental taxes, one can cite several ones. One of them is their potential to reduce the environmental damage through the reduction of greenhouse gases and of local air or water pollution.

Another one is their economic efficiency, in the context of an increasing need of states and governments for new sources of public tax revenues.

In the past, the environmental policy was usually accomplished through command-and-control measures but these approaches proved to be costly and highly targeted (e.g., technology or emissions standards for specific technologies). As such, we assist nowadays to an increasing use of taxes and trades for environmental regulation.

According to OECD [16], imposing a tax directly addresses the market failure caused by the fact that the activities of production or consumption of some goods ignore the environmental harm imposed to others. If this external cost is not taken into account in the market prices, then it is a negative externality and it is considered a market failure. The basic idea is therefore to use taxation to correct negative externalities as pollution and these corrective taxes are often referred as Pigouvian taxation (the concept being introduced by Pigou in 1920). Also, another reason to use taxes is that they give consumers and businesses more flexibility to decide the cheapest way to reduce their environmental footprint. This contrasts with the regulations imposed by the government and specifying exactly how to reduce emissions or with the subsidies and incentives for promoting certain goods practices and favoring only certain ecological solutions. The last two approaches present the disadvantage that they force governments to pick winners and thus to impose an economic strategy and direct the market.

In Europe, as defined in [17], an environmental tax is defined as a tax which base is a physical unit (or an equivalent) of something that has a proven and specific negative impact on the environment, and was established as a tax by ESA (European System of Accounts). ESA divides the environmental taxes into four main categories: energy-related, for transport, pollution and resources.

In practice, the implementation of environmental taxes is challenging and requires to take into account a number of factors when designing them. A list of requirements for a successful design of a green tax was proposed by the OECD and includes the following items:

- The base of an environmental tax should target the pollutant or polluting behavior.
- The scope of an environmental tax should depend on the scope of the damage being addressed
- An environmental taxe should be homogeneous and apply uniformly with only few exceptions
- The tax rate should be proportional with the environmental damage but also reflect the non-environmental externalities and raise revenue.
- The policy for the tax should be credible and its rate predictable such that the public is convinced of its interest and committed to its application.
- The revenue generated by the environmental tax could be used to reduce other taxes or assist fiscal consolidation.
- An environment tax should preserve the competitiveness and allow, if needed, transition periods.

Currently, except for the amount of taxes on motor vehicles, motor vehicle fuels and energy, the rates of environmentally related taxes in OECD countries are too low and, in most cases, below the value of the actual damage. Moreover, they do not bring significant revenues to the governments, with approximately 5% of the total tax revenues in OECD countries (see figure 1 for the revenues in 2000 from motor vehicles and energy taxes report to other taxes by country).

Carbon tax. The largest and the most important potential environmental tax is the carbon (energy) tax. The main purpose of a carbon tax is to reduce the emissions of greenhouse gases which are responsible for global warming and other climate changes. Between these greenhouse gases, the most important one, for which the emissions are easiest to monitor, is the carbon dioxide, emitted primarily through burning fossil fuels. For example, in the USA, in 2013, 94% of CO_2 emissions were from combustion and most of them were coming from large industrial users, such cement factories so there were relatively easy to monitor.

To estimate the carbon tax revenue is therefore relatively straightforward, since it is equal to the tax rate times the emissions subject to the tax. For moderate carbon tax rates, the vast majority of emissions reductions will come from switching from an electricity production based on coal to more renewable energy sources. In several European countries, like France, Denmark or Sweden, this carbon tax has already been put into place but its application is different in each country. In this paper, we present a carbon tax model for industries, taking into account the different fuel sources for the electricity they use, while keeping private the amount of emissions they produced.

4. System architecture overview

This section is dedicated to a more detailed view of the global architecture setting up the environmental tax service for smart industries while ensuring their data privacy.

We consider the following entities:

- Smart factory. This entity is the owner of some private data related to its greenhouse emissions, water waste or to its physical production. To comply with the authorities policies and/or to improve its public image and become popular with its consumers, it is willing to pay environmental taxes, such as the carbon tax. However, for different purposes (fear of concurrency, refuse to release its industrial secrets, planned strategic investments, etc.) it does not want to share the details of its production or its fuels consumption, since the later could be used to deduct its activities.
- Tax service entity. This actor is the one in charge of the service proposing the taxation service. Using functional encryption, it is willing to keep private the data on which the tax model applies and only have access to the final result, i.e. the amount of tax the users of the proposed service have to pay. It is the one which is updated with the latest information necessary to apply the tax (e.g. for the carbon

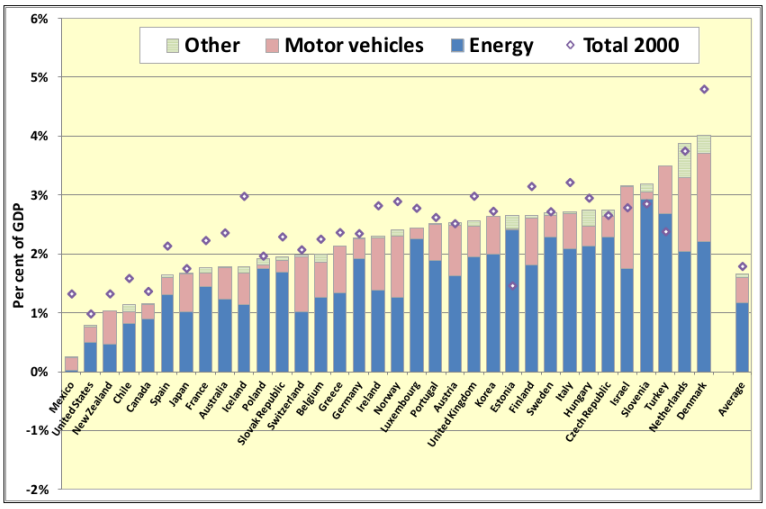

Figure 1. Revenues of the environmental taxes in percentage of GDP per country [16]

tax, the values to use for each emission factor). In practice, this actor can be associated with a local collectivity or a town hall. The taxation service can be used for the payment of the environmental tax on a regular basis (weekly, monthly or annually). Also, it can be proposed by the authorities as a simulator for the industrials such that they can have an idea on the amount of the taxes they have to pay and adapt their environmental strategy or modify their production or their equipments in order to reduce the pollution.

- Qualified Authority. This party is the one setting up the whole taxation strategy and the encryption key manager. It can be associated with national authorities such as governments or the ministry of ecology driving the application of the taxation on the entire territory. Even if responsible in setting up the national tax prices, it delegates the tax collection to the tax service entity.

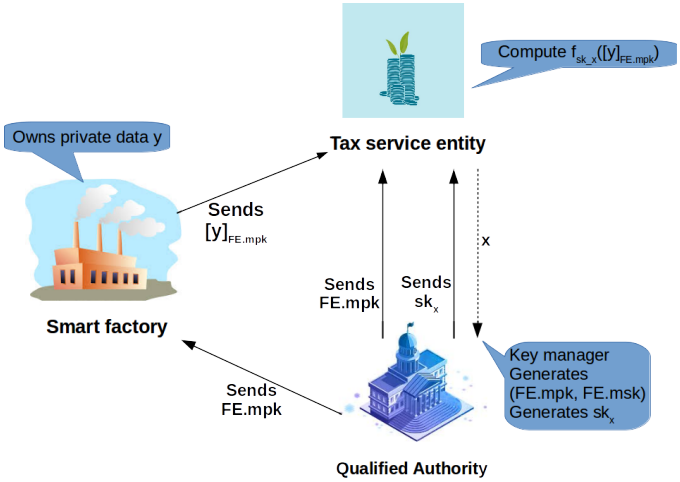

Figure 2. Overall architecture summary

Figure 2 shows the overall architecture. Let us describe more in details this architecture, in which we do not make any assumption (for now) on the underlying functional encryption scheme.

The qualified authority is in charge of the generation and management of the functional encryption keys: the master public and master secret keys (mpk, msk) as well as the secret key sk_x associated with a vector x. For a carbon tax, this vector x corresponds to the emissions factors for each fuel source (coal, gaz, etc.) used in the production of the goods and which are applied in the computation of the carbon dioxide emissions. In the case of local environmental taxes (e.g. local air pollutants or water waste), in which the values of this vector x can be set up locally by each tax service entity, this vector x has to be send to the key manager in order to be applied in the tax model, i.e. to have a secret key associated to it. If the values of the vector x are set up at the national level, the qualified authority will have only to generate sk_x.

On the other side, the smart factory wants to protect its private data, expressed in the form of a vector y, which can consist of its emissions of carbon dioxide by fuel source or other sensitive information on the production. Once it receives the master public key from the qualified authority, it encrypts its data and sends it to the entity in charge of applying the tax. As such, it is sure that not only its data is secure during the exchange but also it stays encrypted on the tax service side.

Finally, the tax service entity, when it receives an encrypted data $[y]$ from the smart factory, applies the tax model expressed as a function f, by using the secret key sk_x. At the end, the tax service entity has access to the tax amount the smart factory has to pay for its environmental

impact. It can cash it right away, store it locally and/or send it to the qualified authority. In the case of the carbon tax, this function f can be expressed as a simple dot product between the encrypted y and sk_x. A functional scheme for inner products such as the one from [1] is therefore enough to implement the taxation on the carbon dioxide. For more complicated taxations models, involving quadratic functions, other, more powerful functional cryptosystems have to be used, as for example [15], or have to be linearized (per se IPFE allows to evaluate higher-degree fonction to the condition that higher-degree terms are encrypted).

5. IPFE constructs

5.1. FE for Inner Products over DDH assumption

In this section we instantiate the scheme of Agrawal et al. [1] in the cyclic group $\mathbb{Z}_p^*$ relying on the standard DDH assumption.

5.1.1. Initialization. Choose a safe-prime p as well as two generators g and h of $\mathbb{Z}_p^*$. Let ℓ denote the dimension of the vectors on which the dot-product will be performed.

Then 2ℓ values are uniformly drawn in $\mathbb{Z}_p$, denoted s_i, t_i ($i \in \{1, ..., \ell\}$), and for each such pair we compute $h_i = g^{s_i} h^{t_i} \mod p$.

The s_i, t_i's define the *master secret key*, msk.

p, g, h and the h_i's define the *master public key*, mpk.

5.1.2. Decryption key generation. Let $x \in \mathbb{Z}_p^\ell$, with the knowledge of msk one can compute

$$s_x = \sum_{i=1}^{\ell} s_i x_i \mod p-1$$

and

$$t_x = \sum_{i=1}^{\ell} t_i x_i \mod p-1.$$

These two values define sk_x, the private secret decryption key associated to vector x.

5.1.3. Encryption of a vector. Let $y \in \mathbb{Z}_p^\ell$ and r uniformly drawn in $\mathbb{Z}_p$. Using mpk, one can compute

- $C = g^r \mod p$.
- $D = h^r \mod p$.
- As well as $E_i = g^{y_i} h_i^r \mod p$, for i from 1 to ℓ.

C, D and the E_i's define the *encryption* of vector y.

5.1.4. Dot-product decryption. Given an encryption of vector y and with the knowledge of both mpk and sk_x, one can compute,

$$E_x = \left(\prod_{i=1}^{\ell} E_i^{x_i} \right) (C^{s_x} D^{t_x})^{-1} \mod p. \qquad (1)$$

It is then easy to verify that $E_x = g^{\sum_i x_i y_i}$.

The last subtlety is that getting the dot-product requires solving a discrete-log problem which hopefully is not too hard when a bound on $\sum_i x_i y_i$ is known and small enough.

5.2. FE for Inner Products over DCR assumption

The main drawback of the version based on DDH assumption lies in the fact that the decryption method is quite expensive. Another option proposed in [1] is to use a solution based on Paillier's composite residuosity assumption and the following property:

For a RSA modulus RSA $N = pq$, the multiplicative group $\mathbb{Z}_{N^2}^*$ has a subgroup of order N generated by $N+1$ in which the discrete logarithm problem is easy to solve.

In the original paper, there are two constructions based on Paillier: a first one, allowing to compute inner products over $\mathbb{Z}$ which requires a prior knowledge on the upper bound for the vectors on which the inner product is securely computed; a second construction, computing inner products over $\mathbb{Z}$ modulo N, in which the setup of master keys does not need the previously upper bound but which uses a stateful key generation algorithm.

In the following, we note y the vector to be encrypted and x the vector associated to a particular secret key.

5.2.1. Initialization. For this particular scheme, we need to know l, the length of x and, respectively, y as well as their infinite norms bounded by X and, respectively, Y. Two safe prime numbers $p = 2 * p' + 1$ and $q = 2 * q' + 1$ are chosen such that $p', q' > 2^{l(\lambda)}$, with l the length of the vectors and λ the security level required. The RSA moduls $N = p * q$ should be superior to $X * Y$ with $X \geq \|x\|_\infty$ et $Y \geq \|y\|_\infty$.

We sample uniformly $g' \hookleftarrow \mathbb{Z}_{N^2}$ and we obtain the subgroup of residues $\mathbb{Z}_{N^2}$ generated by $g = g'^{2N} \mod N^2$.

We sample the l integer values of the vector s according to a discrete Gaussian distribution $D_{\mathbb{Z}^l,\sigma}$ with $\sigma > \sqrt{\lambda} N^{5/2}$ and we compute the terms $h_i = g^{s_i} \mod N^2$.

As such, the master public key mpk is defined by the parameters N, g, the terms h_i for $i = \{1, \ldots, l\}$ and the bound Y. The master secret key msk is composed by the vector s of l-length l and by the upper bound X.

5.2.2. Secret key generation. To generate a key sk associated to a vector $x = (x_1, \ldots, x_l)$, we compute:

$$sk_x = \sum_{i=1}^{l} s_i * x_i.$$

5.2.3. Encryption of a vector. We sample uniformly $r \hookleftarrow \{0, \ldots, \lfloor N/4 \rfloor\}$.

We compute:

$$C_0 = g^r \mod N^2,$$

$$C_i = (1 + y_i N) h_i^r \mod N^2,$$

for $i = \{1, \ldots, l\}$.

The ciphertext for y is $C_y = (C_0, C_1, \ldots, C_l)$.

5.2.4. Decryption and inner product. From the ciphertext C_y and, using the secret key $sk_x \in \mathbb{Z}$, we obtain:

$$C_x = \prod_{i=1}^{l} C_i^{x_i} * C_0^{-sk_x} \mod N^2.$$

We can then use the property stated previously to obtain more easily the value of the discrete logarithm. We thus compute:

$$\log_{1+N}(C_x) = \frac{C_x - 1 \mod N^2}{N}.$$

6. Experimental results

All the experiments were realized using a standard workstation, with a processor Intel Core I7 at 2.6 GHz, with 16 GB of RAM memory and Ubuntu 16.04 as operating system (on 64 bits).

The performances tests use a Sage implementation of IPFE scheme from DCR assumption (see Section 5.2), and, respectively, a Java implementation for the scheme under DDH assumption, described in Section 5.1.

We performed two types of experiments. The first one, based on synthetic data, was used in order to measure the performances in terms of execution time and ciphertext size of the implementation of IPFE scheme. The second one is based on real data and shows the performances when using IPFE for a carbone taxation model.

Table 1 shows the results, for a 128-bit equivalent level of security, obtained when varying the length of the vectors x and respectively y on which the inner products is evaluated. The elements of vectors x and y are integer random numbers in $[0, 1000]$. The columns "setup", "key", "enc" and "dec" show the average execution times (in seconds) for 30 instances for the generation of master keys, the generation of the secret key, the encryption of y and, lastly, the decryption resulting in the inner product. Note that for the DDH case, the decryption time includes the loading of a lookup table used to quickly solve the final discrete logarithm (with a prior knowledge of an upper bound on the resulting value for the inner-product). Column "size" shows the size in kBytes required for a ciphertext in function of the length of the vector to be encrypted. As expected, the times of the setup, encryption, decryption as well as the size of the ciphertexts increase linearly with the length of the original vectors x and, respectively, y. The time for the generation of the secret key from the vector x remains negligible, which seems normal since it requires only basic operations compared with the other functions, using heavier computations like modular exponentiations.

As for the second test, in which we applied IPFE for the computation of a carbon tax for smart factories in a real setting, we performed two types of tax modeling. In a first case, the elements of the vector x which serve as the secret key are expressed as the global coefficients of each electricity source as defined by the International Energy Agency (see Table 2, column "Global"). In the second tax model, we used a finer granularity and consider the values for each source of energy by the technology used (see Table 2, column "Technology").

Since we did not have access to datasets detailing the real consumption of a smart factory, we had to make some assumptions.

First at all, as data sources we make use of the total energy consumed and produced in France in 2015 as reported by RTE (the French transmission system operator) site [18]. The online web service of RTE gives not only the total energy consumption in France in MW at 30 minutes rate but also the detailed production in oil, gas, coal and bioenergy in MW. We make the assumption that the energy production by energy source is equivalent with the energy consumption (we ignore the exchanges with other countries).

Secondly, the heavy industries with the highest CO_2 emissions have to respect an emissions trading scheme. So here we apply the tax model only to SME/SMI type of businesses, directly connected to the distribution grid, which, accordingly to RTE, made up 12% of the net energy consumption in 2015. Finally, there were approximately 138000 SMEs/SMI in France in 2013 [19] and we suppose that their number remains relatively stable over the years. Thus, we apply the carbon tax for an average energy consumer of type SME/SMI in France in 2015. Finally, we consider a taxation with 49 euros per ton of CO_2, considered the amount required for France to reduce its carbon emissions of 20% until 2020 [20].

Table 3 shows the results obtained when applying the IPFE scheme under DCR assumptions to compute the total amount of the carbon tax while protecting the data of an average SME (i.e., its specific energy emissions). For the first test, we use the global coefficients for each energy source and we round it by 100, while for the second, we use more detailed coefficients, rounded to 1000. The consumptions of the SME decomposed by the primary energy source used are rounded each time by a factor of 10000 and encrypted using IPFE under DCR assumption. The setup, key generation, encryption and decryption times are given in their associate columns in seconds, the column "emissions" corresponds to the carbon emissions in tons/MWH and the final column shows the amount of the tax in euros an average SME should pay for a year. The difference from the two amounts comes from taking a finer granularity to describe the enterprise consumption in the second data set.

7. Conclusion and future work

This paper presents a taxation model using functional encryption to ensure data privacy. We show that for the computation of a carbon tax, just an inner-product functional encryption scheme is sufficient since the model is linear. After the description of the architectural framework and a preliminary analysis of a number of threats, we presented the performance results obtained with two instantiations of the IPFE scheme from [1], under different security assumptions. These first experimental tests using both synthetic and real data are very promising, since the performances in terms of

sec	length	setup	key_gen	enc	dec	size (kB)
DCR	10	25.543	0.000	0.507	0.009	6.9
	50	27.554	0.000	2.225	0.102	31
	100	31.448	0.001	0.441	0.134	60.2
	500	67.271	0.004	21.656	0.849	292.5
	1,000	116.076	0.009	45.888	2.985	583.1
DDH	10	1.122	0.186	0.916	0.415	3.8
	50	3.204	0.221	1.927	0.629	16.1
	100	5.523	0.204	3.148	0.678	31.0
	500	25.875	0.397	12.155	1.719	147.6
	1,000	47.294	0.303	23.026	6.591	293.0

TABLE 1. RESULTS FOR IPFE UNDER DCR ASSUMPTION (TOP) AND DDH ASSUMPTION (BOTTOM) WHEN VARYING THE VECTORS DIMENSION.

Source	Technology	Values	Global
Gas	TAC	0.593	0.46
	Co-generation	0.350	
	CCG	0.359	
	Other	0.552	
Oil	CAT	0.777	0.67
	Co-generation	0.459	
	Other	0.783	
Coal	Coal	0.956	0.96
Bioenergy	Waste	0.983	0.98
	Biomass	0.983	
	Biogas	0.983	

TABLE 2. COEFFICIENTS FOR CARBONE EXPRESSED IN T/MWH BY PRIMARY ENERGY SOURCE [21].

l	setup	key	enc	dec	emissions	tax (e)
4	14.557	0	0.202	0.088	24,317	1191.54
11	23.91	0	0.48	0.088	22.113	1083,57

TABLE 3. RESULTS FOR THE CARBON TAX ON AN SME TYPE OF BUSINESS IN FRANCE USING IPFE.

execution times and memory requirements are acceptable, for a standard security level.

Of course, this is just a first proposal of using IPFE in the context of an application for the smart factory and thus many perspectives could be imagined. First at all, the study we conducted on the bibliography related to behavior-based taxation models revealed that the majority of these models are linear ones (or linear by interval). Therefore, the approach presented here could be easily applicable for the safe computation of other linear (or low-degree) taxes.

Secondly, since functional encryption is still at its beginnings, many open questions remain. One of the greatest challenges is to construct secure and efficient functional encryption schemes for richer classes of functions. Still, even with the restricted primitives available today, FE can already be useful in many practical scenarios (e.g. privacy-preserving data search or machine learning).

Acknowledgments

This work was done as part of IRT SystemX projects SCE (Smart City Energy Analytics) and, for the second author, EIC (Environment for Cybersecurity Interoperability and Integration).

References

[1] S. Agrawal, B. Libert, and D. Stehle, "Fully secure functional encryption for inner products, from standard assumptions," Cryptology ePrint Archive, Report 2015/608, 2015, https://eprint.iacr.org/2015/608.

[2] A. C. Yao, "Protocols for secure computations," in *Proceedings of the 23rd Annual Symposium on Foundations of Computer Science*, ser. SFCS '82, 1982, pp. 160–164. [Online]. Available: http://dx.doi.org/10.1109/SFCS.1982.88

[3] O. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game," in *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, ser. STOC '87. ACM, 1987, pp. 218–229. [Online]. Available: http://doi.acm.org/10.1145/28395.28420

[4] C. Gentry *et al.*, "Fully homomorphic encryption using ideal lattices." in *STOC*, vol. 9, no. 2009, 2009, pp. 169–178.

[5] Z. Brakerski and V. Vaikuntanathan, "Fully Homomorphic Encryption from Ring-LWE and Security for Key Dependent Messages." in *CRYPTO*, ser. Lecture Notes in Computer Science, vol. 6841. Springer, 2011, pp. 505–524.

[6] S. Agrawal, D. Boneh, and X. Boyen, "Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical ibe," in *Advances in Cryptology – CRYPTO 2010*, T. Rabin, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 98–115.

[7] D. Boneh and M. K. Franklin, "Identity-based encryption from the weil pairing," in *Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology*, ser. CRYPTO '01. Springer-Verlag, 2001, pp. 213–229. [Online]. Available: http://dl.acm.org/citation.cfm?id=646766.704155

[8] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Proceedings of the 24th Annual International Conference on Theory and Applications of Cryptographic Techniques*, ser. EUROCRYPT'05. Berlin, Heidelberg: Springer-Verlag, 2005, pp. 457–473.

[9] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in *Proceedings of the 14th International Conference on Practice and Theory in Public Key Cryptography Conference on Public Key Cryptography*, ser. PKC'11. Springer-Verlag, 2011, pp. 53–70. [Online]. Available: http://dl.acm.org/citation.cfm?id=1964658.1964664

[10] Z. Brakerski and V. Vaikuntanathan, "Circuit-abe from lwe: Unbounded attributes and semi-adaptive security," Cryptology ePrint Archive, Report 2016/118, 2016, https://eprint.iacr.org/2016/118.

[11] J. Katz, A. Sahai, and B. Waters, "Predicate encryption supporting disjunctions, polynomial equations, and inner products," Cryptology ePrint Archive, Report 2007/404, 2007, https://eprint.iacr.org/2007/404.

[12] R. Gay, P. Méaux, and H. Wee, "Predicate encryption for multi-dimensional range queries from lattices," in *Public Key Cryptography*, ser. Lecture Notes in Computer Science, vol. 9020. Springer, 2015, pp. 752–776.

[13] A. Fiat and M. Naor, "Broadcast encryption," in *Proceedings of the 13th Annual International Cryptology Conference on Advances in Cryptology*, ser. CRYPTO '93, 1994, pp. 480–491.

[14] M. Abdalla, F. Bourse, A. De Caro, and D. Pointcheval, "Simple functional encryption schemes for inner products," Cryptology ePrint Archive, Report 2015/017, 2015, https://eprint.iacr.org/2015/017.

[15] C. Baltico, D. Catalano, D. Fiore, and R. Gay, "Practical functional encryption for quadratic functions with applications to predicate encryption," Cryptology ePrint Archive, Report 2017/151, 2017, https://eprint.iacr.org/2017/151.

[16] O. f. E. C.-o. OECD and Development, "Environmental taxation : A guide for policy makers," Tech. Rep., 2011, https://www.oecd.org/env/tools-evaluation/48164926.pdf.

[17] E. U. EU, *Regulation (EU) No 691/20177 on European environmental accounts*, Std., 2011, http://eur-lex.europa.eu/LexUriServ/-LexUriServ.do?uri=OJ:L:2011:192:0001:0016:EN:PDF.

[18] RTE, "Rte eco2mix annuel definitif 2015 - france energy consumption," 2017, http://www.rte-france.com/fr/eco2mix/eco2mix-telechargement.

[19] INSEE, "Insee statistiques," 2017, https://www.insee.fr/fr/statistiques/1379705.

[20] Coe-Rexecode, "Institut privé d'études économiques," 2017, http://www.coe-rexecode.fr/public/Analyses-et-previsions/Documents-de-travail/Taxe-carbone-opportunites-et-couts-pour-l-economie-et-les-entreprises-francaises.

[21] IEA, "International energy agency," 2017, http://www.iea.org/-bookshop/757-CO2_Emissions_from_Fuel_Combustion_2017.