Multi-Resolution Load Curve Representation with Privacy-preserving Aggregation

Dominik Engel and Günther Eibl
Josef Ressel Center for User-Centric Smart Grid Privacy, Security and Control
Salzburg University of Applied Sciences
Urstein Sued 1, A–5412 Urstein/Salzburg, Austria
Email: {dominik.engel, guenther.eibl}@en-trust.at

***Abstract*—The availability of individual load curves per household in the smart grid end-user domain combined with non-intrusive load monitoring to infer personal data from these load curves has led to privacy concerns. Two types of approaches show high potential to resolve this issue: (i) secure aggregation and (ii) multi-resolution representation with conditional access. In this paper a combination of these two principle approaches is proposed. It is shown formally that secure aggregation and wavelet-based multi-resolution representation are compatible. Furthermore, it is shown that that the wavelet transformation is compatible with existing privacy-preserving protocols and can be used to extend them with additional degrees of freedom. An implementation of the proposed approach is used for evaluation of feasibility in a low-cost embedded environment.**

I. Introduction

Intelligent energy systems, so-called smart grids, revolutionize existing energy grids by combining them with information and communication technology. Smart grids demand accurate and fine-grained data on network status. The widespread roll-out of smart meters is one of the consequences. Smart meters record energy consumption in a specified granularity (usually the time between readings is between 1 second and 15 minutes) and have the ability to transmit these load curves in a specified interval (e.g., once a day).

It has been shown that personal information on the end-user can be inferred from fine-grained load curves [1], [2], and this has led to privacy concerns [3], [4]. The accuracy of the inferred information is directly connected to the available resolution of the load data. A number of methods have been proposed to balance the need for privacy with the information needed for correct operation of smart grids. Two types of approaches show high potential to resolve this issue: (i) secure aggregation of encrypted load curves, and (ii) representation of load curves in multiple resolutions, each associated with different access levels.

Approaches of the first type can again be divided into two categories: protocols using masking [5], [6] and protocols using homomorphic encryption. In this paper the focus is put on the second kind of protocols. Privacy-enabling encryption for smart meter data by the use of homomorphic encryption is suggested by, e.g., [7]–[10], allowing the aggregation of encrypted signals, also termed "secure signal processing". A recent overview of secure signal processing, covering four proposals for privacy-preserving smart metering aggregation is given by [11].

Approaches of the second type suggest to represent load curve data in multiple resolutions, where each resolution can be used for a different purpose, e.g., low resolution for billing, and is therefore disclosed to selected parties only, e.g., [12]. Using the wavelet transform to produce an integrated bitstream supporting multiple resolutions has been proposed by [13]. Combined with conditional access, i.e., different encryption keys for each resolution, this wavelet-based representation allows user-centric privacy management: access can be granted or revoked for each resolution. Access to high resolutions, which are privacy-sensitive, may be reserved to a small number of trusted entities only, whereas resolutions of medium granularity may be provided more freely, e.g., to contribute to network stability (in exchange for lower energy prices or other incentives).

In this paper, a privacy-preserving smart metering method that combines the two types of approaches, namely homomorphic encryption and multi-resolution representation, is proposed. This enhances the possibilities for managing privacy requirements, as the combination of both methods significantly increases the degrees of freedom. Access control does not relate to the aggregated signal as a whole anymore, but access can be granted on the aggregate on each resolution *individually*. This is an important feature, as it allows to grant access to participants in the smart grid system, based on their roles and the functions they have to fulfill. Each role can be assigned access to the aggregate on the minimum resolution necessary to fulfill the functions associated with this role.

The rest of this paper is structured as follows. Section II summarizes the principle of wavelet-based load curve representation and homomorphic encryption. In Section III the combination of the two approaches is introduced and their compatibility is proven mathematically. Results are discussed in Section IV-A. The usability of the wavelet transformation with existing protocols is discussed in IV-B. Section V concludes the paper.

II. Background

A. Wavelet-based Representation

A wavelet transform starts with the original load curve and is recursively performed in S steps. In each step s half of the

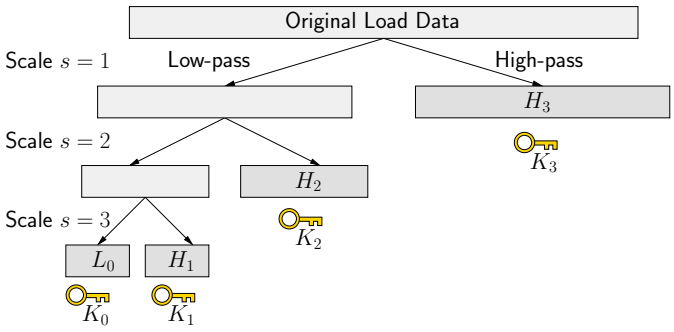

Fig. 1. Wavelet transformation and encryption of load curve

data (the highpass data) $\tilde{H}_s$ are remembered as the wavelet coefficients (subband) of scale s and the next step is performed for the lowpass data. At the end of the transformation the final subband $\tilde{H}_S$ consists of 2^{-S} samples. The higher the scale s, the lower the time resolution $r := S - s$. Reindexing $H_r = \tilde{H}_{S-s}$, at the end of the transformation one obtains a sequence $L_0,\ H_1, \ldots,\ H_S$, see Fig. 1.

The synthesis step of the inverse wavelet transform W^{-1} starts with the lowest resolution $r = 0$. To get the next higher resolution of the signal the next higher resolution subband is needed, so that in a series of S steps one finally obtains the original load curve (since we only consider lossless transformations). In order to provide a signal m_R with maximum resolution R only R synthesis steps must be performed and only the subbands with resolution $r \leq R$, i.e., $L_0,\ H_1, \ldots,\ H_R$ are needed. Denoting the selection of the R highest resolutions as T_R this can be written as

$$m_R = W^{-1}\left[T_R(W[m])\right] \tag{1}$$

Making the signal available at the needed resolution instead of the full resolution increases privacy because less (personal) information can be deduced. For allowing differentiated access control, each subband (i.e., each resolution) of the resulting wavelet decomposition is encrypted with a different key, as illustrated by Fig. 1. A combination with public key infrastructures to allow fine-grained access control is possible. A hierarchical key creation scheme can be used to minimize overhead for key exchange. The result of this process is an encrypted bitstream that forms an integrated representation of all resolutions. Note that no data expansion occurs, i.e., the size of the final bitstream equals the size of the original data.

The operator T_R can be generalized to be any transformation T of the wavelet coefficients to be used for example for denoising. In the simplest case, using a global threshold η it could be defined as $T(W[m]) = W[m]\delta(m-\eta)$ with δ denoting Dirac's delta function, for more sophisticated denoising methods, see [14]. Using denoising transformations could turn out to be valuable for transmission of signal aggregations.

Wavelets are also used for the generation of features in load forecasting [15], [16]. The representation suggested in this paper may be of advantage for load forecasting methods based on wavelets. However, the main focus of this paper is load aggregation with access control to different resolutions.

Key generation: generate public keys g and n and private key λ - Generate private key prime numbers p and q randomly - Set private $\lambda = \mathrm{lcm}(p-1, q-1)$ - Set public key $n := pq$ - Select public key g with the property $\gcd(L(g^\lambda \mod n^2),\ n) = 1$ for $L(u) := \frac{u-1}{n}$
Encryption: given message $m \in \mathbb{Z}_n$ - Generate random number $r \in \mathbb{Z}_n^*$ - Ciphertext $c = E(m; g, n) = g^m \cdot r^n \mod n^2 \in \mathbb{Z}_{n^2}^*$
Decryption: given ciphertext $c \in \mathbb{Z}_{n^2}^*$ - $m = D(c; g, n, \lambda) = \frac{L(c^\lambda \mod n^2)}{L(g^\lambda \mod n^2)} \mod n \in \mathbb{Z}_n$

TABLE I
HOMOMORPHIC ENCRYPTION

B. Homomorphic Encryption

Following previous proposals [7]–[9], a Paillier cryptosystem [17] is employed. The whole encryption and decryption process can be split into three parts: key-generation, encryption and decryption. It is described in Table I. Note that the numbers g, n and λ are kept fixed and are omitted for simplicity.

Homomorphic encryption has the following important property, which is called the *additive property*:

$$D\left(E(m_1)E(m_2) \bmod n^2\right) = m_1 + m_2 \bmod n. \tag{2}$$

This property means that the decryption of the product of the *ciphertexts* is the sum of the original plaintext messages.

C. Privacy Preserving Protocols

In [7]–[10], protocols using homomorphic encryption are proposed as tools for privacy conserving aggregation of load curves. As it is done there, the smart grid network considered consists of N households each having one smart meter installed and an aggregator (Fig. 2).

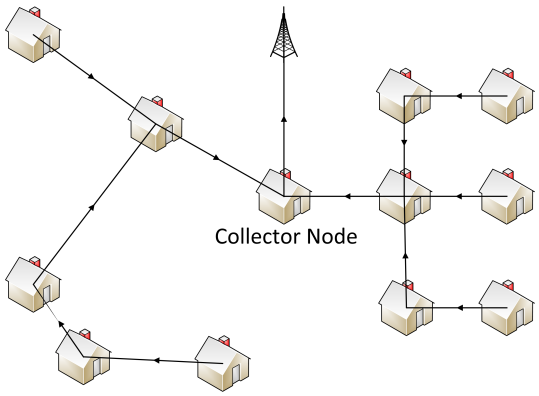

Fig. 2. Aggregation of encrypted signals

The network is assumed to have tree-like connections. Each smart meter i sends its measured load m_i in encrypted form to its parent smart meter. The parent smart meter multiplies the obtained encrypted signals with its own encrypted signal and in turn sends this product to its parent node. Finally, the aggregator multiplies the obtained signals and decrypts the product. Due to the homomorphic property, the result is the

sum of the measured loads. With E and D denoting Pailler encryption and decryption this can be stated as

$$D\left(\prod_i E(m_i) \bmod n^2\right) = \sum_i m_i \mod n. \quad (3)$$

Privacy is preserved because of the distributed way of processing. Smart meters only have the plaintext information of their own messages, because they cannot decrypt the messages they get. The aggregator can decrypt messages, but, as it receives the product of the individual ciphertexts, can only decrypt the sum of the load curves.

III. Aggregation of encrypted wavelet-transformed signals

The goal of this paper is an extension of the distributed homomorphic encryption process in a way that is compatible with the wavelet transformation. In particular it is shown that when homomorphic encryption is applied to a signal represented in the wavelet domain, homomorphic additivity is not only preserved, but can be separately exploited for each resolution.

In [13], a variety of wavelet filters regarding their utility for the multi-resolution representation of load curves was evaluated. Only lossless transformations are useful in the context of smart metering. The Haar wavelet filter preserves the average over all resolutions, which is an important property for many use cases. Using the lifting implementation of the Haar wavelet, the transformation can be realized efficiently. The lifting steps for the forward transform with the Haar wavelet have been formulated by [18]. As the original Haar wavelet uses real coefficients, it is ill-suited for use with homomorphic encryption. Therefore, for the combination with homomorphic encryption a modified version of the Haar wavelet is used that only produces integer values for the transformed load curve (where $\tilde{L}_0 = X[i]$ is the input signal, $\tilde{H}_s[i]$ and $\tilde{L}_s[i]$ are the resulting high-pass and low-pass subband at scale s, respectively, with i denoting the position within the signal):

$$\tilde{L}^{(0)}_{s+1}[i] = \tilde{L}_s[2i] \quad (4)$$
$$\tilde{H}^{(0)}_{s+1}[i] = \tilde{L}_s[2i+1] \quad (5)$$
$$\tilde{H}_{s+1}[i] = \tilde{H}^{(0)}_{s+1}[i] - \tilde{L}^{(0)}_{s+1}[i] \quad (6)$$
$$\tilde{L}_{s+1}[i] = 2\tilde{L}^{(0)}_{s+1}[i] + \tilde{H}_{s+1}[i]). \quad (7)$$

The inverse transform can be written as:

$$\tilde{L}^{(0)}_s[i] = \frac{1}{2}\tilde{L}_{s+1}[i] - \frac{1}{2}\tilde{H}_{s+1}[i] \quad (8)$$
$$\tilde{H}^{(0)}_s[i] = \tilde{H}_{s+1}[i] + \tilde{L}^{(0)}_s[i] \quad (9)$$
$$\tilde{L}_s[2i+1] = \tilde{H}^{(0)}_s[i] \quad (10)$$
$$\tilde{L}_s[2i] = \tilde{L}^{(0)}_s[i]. \quad (11)$$

Note that the average of the original series is still preserved over all resolutions for the modified Haar filter:

$$\forall s: \sum_i X[i] = 2^{-s} \sum_k \tilde{L}_s[k].$$

Fig. 3 shows the aggregation of a number of multi-resolution load curves at a collector node. Homomorphic encryption is applied to each resolution r separately with a different key $K_r = (g_r,\ n_r)$. The resulting signal m is the sum of all signals m_i (each of which has a maximum resolution of R) at resolution $r \leq R$, whereby W denotes a wavelet transformation. The collector node can perform aggregation (i.e., multiply) in the encrypted domain, i.e., it does not have any keys. This ensures that the aggregator cannot get information about the loads of its children, e.g., by divisions.

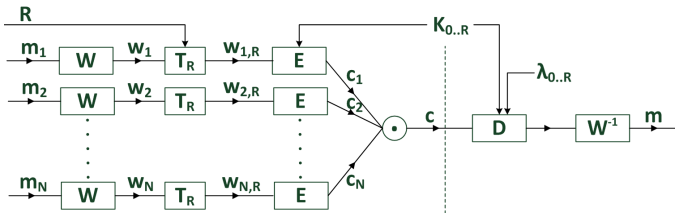

Fig. 3. Aggregation of encrypted multi-resolution load curves

Writing the procedure mathematically yields the following calculation of the ciphertext c

$$c = \prod_i E(T_R(W[m_i])) \bmod n^2.$$

The ciphertext c is decrypted by the aggregator in the following way

$$m = W^{-1}[D(c) \bmod n]$$

Using this procedure the wavelet transformation is compatible with homomorphic encryption, i.e., the homomorphic property that the message m equals the sum of the messages is preserved (choosing R = S). Even more, choosing $R < S$, the decrypted message m equals the sum of the messages of resolution R:

$$m = W^{-1}[\prod_i E(T_R(W[m_i])) \bmod n] = \sum_i m_{R,i} \bmod n. \quad (12)$$

The aggregator gets the product of the encrypted messages and can therefore not extract any information about the individual messages. However, it can calculate the sum of the messages which is the information needed, e.g., for load forecasting. Note again that the product of the ciphertexts is calculated in a distributed way by the smart meters and not by the aggregator. The number n must be chosen big enough so that $\prod_i E(T_R(W[m_i])) < n^2$ and $\prod_i E(T_R(W[m_i])) < n$ hold. For sake of readability the modulo parts of the calculations are therefore omitted in the proof.

Proof: Without loss of generality two messages are considered. To simplify the analysis the notation $y_i := T_R(W[m_i])$ is used, so $E(T_R(W[m_i])) = E(y_i)$. The aggregator calculates the signal $W^{-1}[D(c)]$. Using the fact that the ciphertext c is the product of the individual ciphertexts and

the homomorphic encryption property leads to

$$\begin{aligned} W^{-1}[D(c)] &= W^{-1}[D(c_1 c_2)] \\ &= W^{-1}[D(E(y_1)E(y_1))] \\ &= W^{-1}[y_1 + y_2] \end{aligned}$$

Substituting for the y_i, using the linearity of the wavelet transform and the definition of m_R yields

$$\begin{aligned} W^{-1}[D(c)] &= W^{-1}[T_R(W[m_1]) + T_R(W[m_2])] \\ &= W^{-1}[T_R(W[m_1])] + W^{-1}[T_R(W[m_2])] \\ &= m_{R,1} + m_{R,2} \end{aligned}$$

So in general for I different messages and ciphertext $c = \prod_i c_i$ the desired property (12)

$$W^{-1}[D(c)] = \sum_{i=1}^{I} m_{R,i} \tag{13}$$

is obtained. ■

An example use-case scenario is the use of aggregated load information for energy monitoring by the network operator, as, e.g., suggested by [11]. The approach proposed here adds an additional layer of flexibility by making the aggregates available at different resolutions with access being granted to parties on the resolutions with the necessary granularity to fulfill a specific task. In combination with suitable key management, this approach implements the "need-to-know" principle of access for aggregated signals.

IV. Results

A. Cost and complexity

The proposed method has been implemented as a proof of concept in Java (Oracle Java v8 preview with ARM-extensions) and evaluated in a low-cost ARM-based environment (Beagleboard BB-XM-00, Rev C, with a TI DM3730 1Ghz ARM processor and 512MB of RAM) running Ubuntu Linux 12.04.

Results are shown in Table II: Each value represents the execution time for a single load curve consisting of 96 values for the wavelet transform combined with different encryption settings, averaged over 400 load curves with 100 encryptions each (acquisition of the load curve and key generated are not considered in the timing results). WAV denotes the wavelet transform only, without any encryption applied. AES denotes the wavelet transform followed by encryption with the symmetric AES cipher with a 128 bit key for each subband. HYB denotes hybrid encryption, which adds RSA 2048 bit public key encryption of the AES keys with a different public key for each subband. Finally, PAI-n denotes Pailler encryption with a module of n bits and a different key for each subband.

It can be seen that by using a lifting implementation the transformation is very fast and the computational overhead is negligible compared to the encryption step. Homomorphic encryption comes at the cost of a significant increase in computational overhead compared to conventional encryption.

	WAV	AES	HYB	PAI-256	PAI-512	PAI-1024
Exec. time	**0.15**	1.91	72.4	1,649	11,452	85,355
Std. dev.	**0.01**	0.03	0.1	16	22	133

TABLE II
EXECUTION TIME IN MILLISECONDS FOR TRANSFORMING/ENCRYPTING A SINGLE LOAD CURVE (AVERAGE OVER 400 LOAD CURVES WITH 100 ENCRYPTIONS EACH)

The results show that the computational demands grow exponentially with the module size. Considering that 256 and 512 bit modules will in most use-cases not be sufficient in terms of security, the increased execution time for module sizes that are more secure provides a challenge. While AES encryption only takes 1.9 ms, for the used (non-optimized) implementation, Paillier encryption of a load curve with 96 values takes nearly 90 seconds for a module of 1024 bit. It needs to be pointed out that this drawback also affects all previously proposed methods for homomorphic load curve encryption that rely on a Paillier cryptosystems. Optimization of the implementation is one option to be considered. Another option is to investigate the utility of alternative homomorphic encryption schemes.

The approach proposed here adds the possibilities offered by wavelets to distributed homomorphic encryption and decryption schemes. It is therefore compatible with any homomorphic encryption scheme. The wavelet transformation can be seen as an add-on which is compatible with homomorphic encryption. Since the computational cost of the wavelet transformation is small, the computational cost of the main privacy preserving protocol dominates the overall cost. Thus, the complexity evaluation given in [11] can be used as a complexity assessment for different kinds of privacy preserving protocols.

B. Usability with existing protocols

The extension of the privacy preserving protocol was designed for the protocol used in [7]. Thus it can readily be used within privacy preserving protocols, which directly rely on the homomorphic encryption property such as [7], [9]. Here we study, if wavelets can also be used together with other protocols found in the literature.

The method in [8] combines Paillier's homomorphic encryption with additive secret sharing. Generally, additive masking terms need no adjustment since they cancel out in the decryption step before the inverse transformation takes place. Thus, the method is compatible with the wavelet transformation.

The method in [10] extends [7] by preserving data integrity. The wavelet transformation is compatible with this method since it is mostly based on the ciphertext. There, it is irrelevant if the encrypted message is in its original or in a transformed form. Decryption is only done in the incremental verification process where the compatibility can be verified for each individual step.

Other existing protocols need a homomorphic property but do not use Paillier's homomorphic encryption [6], or they use other principles as for example masking [6], [5]. Next, it will be checked, if the wavelet transformation is also compatible with these methods.

In [6], the modulo operation is used for homomorphic encryption instead of Paillier's homomorphic encryption scheme. Privacy is achieved by masking. The second main feature is the addition of Laplacian noise for differential privacy. This encryption scheme can be made compatible with the wavelet transformation by the following modifications: the multiplication in the aggregation step must be substituted by an addition. The signal $T_R(W(m_i))$ corresponds to the signal X_i in [6]. As already stated above, the additive masking terms need no adjustment. The same argument holds for the keys added for ensuring confidentiality with the aggregator. However, the terms for differential privacy need to be modified. The added noise must be adapted in two ways due to the inverse transformation W^{-1} arising in the decryption step: first, the parameter λ must be chosen suitable for the signals $W^{-1}(T_R(W(m_i))) = m_{R,i}$. Second, the noise added to each signal $T_R(W(m_i))$ which consists of the subtraction of two gamma distributions (with the adapted parameter λ) must be transformed by W which later cancels the W^{-1} in the decryption step. With these changes wavelets are compatible with the method of [6].

In [5], four different protocols which rely on masking are described. These protocols can be categorized into so-called aggregation and comparison protocols. The aggregation protocols are compatible with wavelets. However, in the comparison protocols, the transformed sum of the values is in the exponent of the generating element of the Diffie-Hellman group. As the reverse transformation cannot be calculated, wavelets are not compatible with these comparison protocols.

Summarizing, the wavelet method is compatible with existing privacy preserving protocols except comparison protocols. Adaptations are needed for differential privacy.

V. Conclusion and Outlook

The proposed approach enables access models on a "need-to-know" basis for secure signal processing. This adds flexibility to existing approaches and enhances privacy. Access control to encrypted aggregates is not binary for the whole signal anymore, but instead can be granted to parties for individual resolutions, based on their roles and the associated specific needs in terms of data resolution. A proof has been given that shows that the wavelet transform is compatible with any homomorphic encryption method. Furthermore, the proposed approach can be included in most existing privacy-preserving protocols to enhance the degrees of freedom. Computational demands of homomorphic encryption schemes in general remain a challenge. The overhead for multi-resolution processing is negligible compared to the complexity of encryption.

Like most papers this paper focuses on methodological aspects. In the future we will extend existing work [19] and investigate how these methods can be applied to the relevant use cases like energy feedback, billing or grid stability including practical aspects such as robustness against losing the connection to individual smart meters.

Acknowledgements

The financial support by the Austrian Federal Ministry of Economy, Family and Youth and the Austrian National Foundation for Research, Technology and Development is gratefully acknowledged.

The authors thank company partner Salzburg AG for providing anonymized real-world load curves for testing.

References

[1] G. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the IEEE*, vol. 80, no. 12, pp. 1870–1891, Dec. 1992.

[2] A. Molina-Markham, P. Shenoy, K. Fu, E. Cecchet, and D. Irwin, "Private memoirs of a smart meter," in *Proc. 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building*, ser. BuildSys '10. New York, NY, USA: ACM, 2010, pp. 61–66.

[3] P. McDaniel and S. McLaughlin, "Security and privacy challenges in the smart grid," *IEEE Security Privacy Magazine*, vol. 7, no. 3, pp. 75–77, 2009.

[4] M. Lisovich, D. Mulligan, and S. Wicker, "Inferring personal information from demand-response systems," *IEEE Security & Privacy*, vol. 8, no. 1, pp. 11–20, 2010.

[5] K. Kursawe, G. Danezis, and M. Kohlweiss, "Privacy-friendly aggregation for the smart grid," in *Privacy Enhanced Technology Symposium*, 2011, pp. 175–191.

[6] G. Acs and C. Castelluccia, "I have a dream! (differentially private smart metering)," in *Proc. Information Hiding Conference*, 2011, pp. 118–132.

[7] F. Li, B. Luo, and P. Liu, "Secure information aggregation for smart grids using homomorphic encryption," in *Proc. of First IEEE International Conference on Smart Grid Communications*, Gaithersburg, Maryland, USA, Oct. 2010, pp. 327–332.

[8] F. Garcia and B. Jacobs, "Privacy-friendly energy-metering via homomorphic encryption," in *Security and Trust Management*, ser. Lecture Notes in Computer Science, J. Cuellar, J. Lopez, G. Barthe, and A. Pretschner, Eds. Springer Berlin / Heidelberg, 2011, vol. 6710, pp. 226–238.

[9] Z. Erkin and G. Tsudik, "Private computation of spatial and temporal power consumption with smart meters," in *Proceedings of the 10th international conference on Applied Cryptography and Network Security*, ser. ACNS'12. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 561–577.

[10] F. Li and B. Luo, "Preserving data integrity for smart grid data aggregation," in *Smart Grid Communications (SmartGridComm), 2012 IEEE Third International Conference on*, 2012, pp. 366–371.

[11] Z. Erkin, J. Troncoso-Pastoriza, R. Lagendijk, and F. Perez-Gonzalez, "Privacy-preserving data aggregation in smart metering systems: An overview," *Signal Processing Magazine, IEEE*, vol. 30, no. 2, pp. 75–86, March.

[12] C. Efthymiou and G. Kalogridis, "Smart grid privacy via anonymization of smart metering data," in *Proceedings of First IEEE International Conference on Smart Grid Communications*, Gaithersburg, Maryland, USA, Oct. 2010, pp. 238–243.

[13] D. Engel, "Wavelet-based load profile representation for smart meter privacy," in *Proc. IEEE PES Innovative Smart Grid Technologies (ISGT'13)*, Washington, D.C., USA, Feb. 2013, pp. 1–6.

[14] A. Anestis, J. Bigot, and T. Sapatinas, "Wavelet estimators in nonparametric regression: a comparative simulation study," *Journal of Statistical Software*, vol. 6, pp. 1–83, 2001.

[15] C. Chen, B. Das, and D. J. Cook, "Energy prediction based on resident's activity," in *Proceedings of the 4th International Workshop on Knowledge Discovery from Sensor Data*, Jul. 2010.

[16] C. Guan, P. Luh, L. Michel, Y. Wang, and P. Friedland, "Very short-term load forecasting: Wavelet neural networks with data pre-filtering," *IEEE Transactions on Power Systems*, vol. 28, pp. 30–41, 2013.

[17] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Proceedings of Eurocrypt '99, Advances in Cryptology*, ser. Lecture Notes in Computer Science, J. Stern, Ed., vol. 1592. Prague, Czech Republic: Springer, May 1999, pp. 223–238.

[18] I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *J. Fourier Anal. Appl.*, vol. 4, no. 3, pp. 247–269, 1998.

[19] M. Jawurek, F. Kerschbaum, and G. Danezis, "Privacy technologies for smart grids - a survey of options," Microsoft Research, Tech. Rep., 2012.