# Ensemble Regression for 1-Bit Channel Estimation

Ahmed Elsheikh[1], Ahmed S. Ibrahim[2], and Mahmoud H. Ismail[3]

[1]Department of Eng. Mathematics and Physics, Faculty of Engineering, Cairo University, Giza 12613, Egypt
[2]Electrical and Computer Eng. Dept., Florida International University, Miami, FL 33174, USA
[3]Department of Electrical Engineering, American University of Sharjah, PO Box 26666, Sharjah, UAE
{eng.ama.elsheikh@gmail.com, aibrahim@fiu.edu, mhibrahim@aus.edu}

***Abstract*—Employing 1-bit analog-to-digital converters (ADCs) is necessary for large-bandwidth massive multiple-antenna systems to maintain reasonable power consumption. However, conducting channel estimation with such 1-bit ADCs and with low complexity is a challenging task. In this paper, we propose to employ an Ensemble Regression (ER) model to conduct low-complexity and high-quality channel estimation. The amount of proposed computations are less than $3\%$ of that proposed by similar deep learning (DL) methods, and in turn requires approximately $4\%$ of the power consumed in computations while maintaining the same level of performance.**

***Index Terms*—Analog-to-digital converters, Channel estimation, Ensemble Regression.**

## I. Introduction

Emerging applications such as extended reality (XR) require high data rates, which can be accomplished using massive multiple antenna communication systems over high-bandwidth millimeter wave (mmWave) spectrum band. However, having such large bandwidth and consequently higher sampling frequency require high power consumption from some transceiver components such as analog-to-digital converters (ADCs) [1]. For example, power consumption of ADCs with resolution above 6 bits increases quadratically with the sampling frequency [2]. Such high power consumption led to a research direction of having low-resolution ADCs, with especial emphasis on 1-bit ADCs. This is especially true in massive MIMO system, in which each antenna can be equipped with a 1-bit ADC [3]. However, such low-resolution ADCs encounter challenges regarding accurate *channel estimation*, and this is the *scope* of this paper.

One way to improve channel estimation with low-resolution ADCs is by assuming specific random distributions for the channel models [4]–[7]. As practical channels deviate from standard channel models, such model-based low-resolution channel estimation solutions require long pilot sequences, which hinders their practicality [1]. Machine learning (ML) can address the challenges of such model-based channel estimation, and this is the *motivation* of this paper.

978-1-6654-8237-0/22/$31.00 ©2022 IEEE

A few Deep Learning (DL) models (e.g., [8]) were introduced to have channel estimation, with 1-bit ADC, without assuming a specific form for the channel model. Furthermore in [9], systems with a mix of high- and low-resolution ADCs were employed, but only the received signals from the high-resolution ADCs were used for channel estimation. Channel estimation in OFDM systems, which use 1-bit ADCs, was considered in [10] but only for single-antenna systems. In addition to channel estimation, a DL model was considered in [11] for joint channel estimation and data detection in low-dimensional MIMO systems. Finally, a DL model for data detection alone, using 1-bit ADCs, was introduced in [12].

The common feature among such DL-based channel estimation models (e.g., [8]) is that they all are very computationally demanding, as they require a large number of multiplication operations that grows with the number of neurons composing such DL models. Therefore, there is a need to find low-complexity ML model for low-resolution-ADCs channel estimation, and this is the *goal* of this paper.

The Ensemble Regression (ER) model comes from the family of additive models similar to deep neural networks [13]. Generally, ensemble models depend on adaptively aggregating the approximation of several elemental approximators instead of learning all of the approximations simultaneously as in the case of DL. This enables ER to achieve respectable performance while utilizing simpler elemental approximators. Also, ER achieves such low complexity by requiring lesser model parameters and simpler arithmetic operations to train, which, in turn, means faster training and adaptation, and easier implementation using a combinational logic circuit [14].

In this paper, we propose an ER model for channel estimation with 1-bit ADCs, which is based on regression trees that are essentially zero-order hold approximators. Such ER model aims to accomplish a good tradeoff point between the quality of the estimation and its practicality in terms of the number of required arithmetic operations. To the best of the authors' knowledge, this is the first time to employ ER for channel estimation from quantized 1-bit ADCs, and this is the main *contribution* of

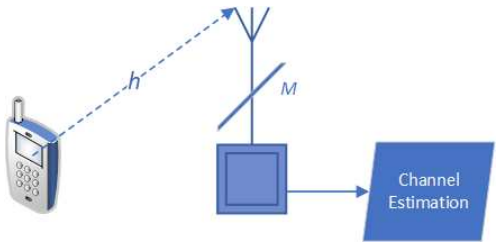


Fig. 1. System model.

this paper. The rest of the paper is organized as follows: in Section II, we present the system model and formulate the 1-bit ADC channel estimation problem. In Section III, we describe the proposed ER model as a part of the Fast 1-Bit Ensemble Regression for Channel Estimations (F1ERCE) approach to solve the problem. In Section IV, we present our numerical results before the paper is finally concluded in Section V.

## II. System Model and Problem Formulation

In this section, we present the system model and problem formulation.

### A. System Model

Fig. 1 shows the system model, which consists of a single base station that is equipped with $M$ antennas and communicates with a single-antenna user equipment (UE). For simplicity of illustration, Fig. 1 only shows a single UE. The receiver chain of the base station is entirely composed of 1-bit ADCs. Additionally, we assume a time division duplexing (TDD) system operation, in which the channel is estimated through uplink training and then employed for downstream data transmission.

We consider a general geometric channel model with $L$ paths that sums up the gains $\{\gamma_l\}_{l=1}^{L}$ of the different array responses $\mathbf{a}(\theta_l)$ arriving with angles $\{\theta_l\}_{l=1}^{L}$ such that $\mathbf{h} = \sum_{l=1}^{L} \gamma_l \mathbf{a}(\theta_l)$.

For the purpose of channel estimation, we assume the UE sends a pilot signal $\mathbf{p}$ of dimension $N \times 1$ that spans $N$ consecutive time slots. Also, let the $M$-antenna channel between the base station and UE be denoted by the $M \times 1$ channel vector $\mathbf{h}$. The received $M \times N$ signal matrix at the base station, over $N$ time slots, can be modeled as $\mathbf{Y} = \mathbf{h}\,\mathbf{p}^T + \mathbf{N}$, where $\mathbf{N}$ is an $M \times N$ additive Gaussian noise matrix and $T$ denotes vector transposition. Utilizing 1-bit ADC of the received signal at each receive antenna, let $\mathbf{X}$ be an $M \times N$ matrix that contains all 1-bit quantized measurements of $\mathbf{Y}$. Knowing $\mathbf{X}$ and $\mathbf{h}$, the base station is able to find an estimate of the $M \times 1$ channel vector, to be denoted as $\hat{\mathbf{h}}$.

As for downlink data transmission, we assume that the base station utilizes transmit beamforming using the estimated 1-bit channel by multiplying the data signals at the $M$ antennas by $\hat{\mathbf{h}}/\|\hat{\mathbf{h}}\|$. As a result the received signal-to-noise-ratio (SNR) per each transmit antenna can be defined as

$$\text{SNR} = \frac{\rho}{M} \frac{|\hat{\mathbf{h}}^H \mathbf{h}|^2}{\|\mathbf{h}\|^2}, \tag{1}$$

where $\rho$ is the mean reception SNR before beamforming. The SNR will be used as the performance measure for the quality of the low-resolution channel estimation.

### B. Problem Formulation

The purpose of this paper is then to examine the construction of an effective channel estimation technique for constructing the channel $\mathbf{h}$ from the highly quantized signal $\mathbf{X}$. Our aim is to construct a channel estimation approach that minimizes the mean-squared error (MSE) between the estimated and original channel vectors which can be defined as

$$\text{MSE} = \mathbb{E}[\|\mathbf{h} - \hat{\mathbf{h}}\|^2], \tag{2}$$

where $\hat{\mathbf{h}}$ is estimated using $\mathbb{E}[\mathbf{f}(\mathbf{x})]$ and $\mathbf{f}(\cdot)$ is a vector function of the low-complexity and power-efficient zero-order-hold regression-trees.

## III. ER for Channel Estimation

In this section, we investigate the proper usage of another additive model, namely the ER, to achieve the same task but with significantly lesser computations.

The ER learner is composed of weak regression models, namely regression tree models that are trained to collaboratively learn a more complex model. This is analogous to the neural network that uses weak neurons and augment their regression power to achieve highly complex regressions. The ensemble learning technique used is the Least-Square Boosting (LSBoost) [15] and it is described in Algorithm 1 as a part of the proposed F1ERCE approach.

The building block, which is the tree regression simply does a zero-order piece-wise regression for a given set of observations. The piece-wise, splits are chosen based on a greedy algorithm that searches for the split that minimizes the MSE. The ensemble learning comes from giving more attention to the residuals of the estimates of the ensemble when learning new trees. Finally, the ensemble decision is a weighted average of the estimations of all trees.

For each value of the output channel dimensions, an ensemble is learnt to map the inputs to one of the desired output dimensions. For example, if there are 10 antennas, each will have a complex channel estimate, and hence we will need 20 ER models.

Fig. 2 shows a simple regression example where the estimations of $T$ simple regression trees are combined using the weights $\eta$ to produce the final estimation.

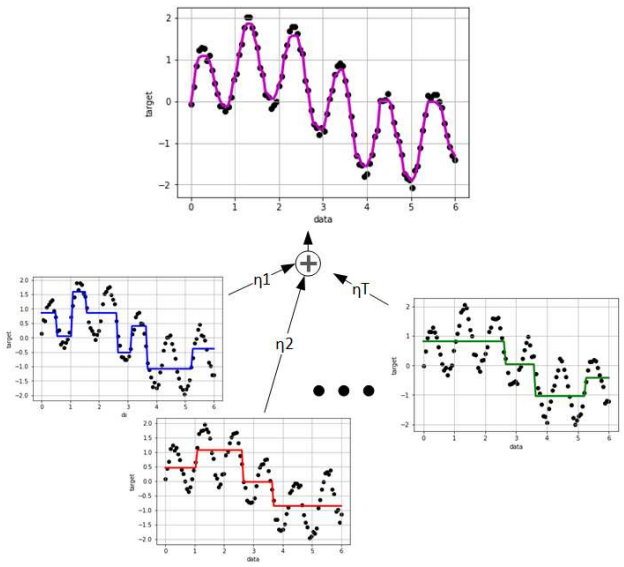


Fig. 2. Illustration for the Ensemble Regression model.

### A. The Proposed F1ERCE Algorithm

The first step is to prepare the training data to be ready for modelling. That is to vectorize each observation of the measurements matrices $\mathbf{X}$. This operation produces a row vector $\mathbf{z}$ of dimensions $1 \times 2MN$, which is the total number of elements in the measurement matrix with the real and imaginary parts concatenated together. The matrix containing all training vectors $\mathbf{Z}$ would then have a dimensionality of $D \times 2MN$ where $D$ is the number of observations in the training set. Then, the proposed F1ERCE algorithm compresses the dimensionality of $\mathbf{Z}$ using Principal Components Analysis (PCA). This projection uses the $K$ first eigenvectors of the correlation matrix of the training data $\mathbf{Z}$ to reduce the dimensionality of the input while retaining a significant amount of the carried information. This compressed version of the input $\mathbf{Z}_K$, which has a dimensionality of $D \times K$, helps in reducing the number of learning parameters required later on by the ER model.

Finally, the channel estimate for a given compressed signal observation denoted by $\mathbf{z}_d$, which is a row vector in matrix $\mathbf{Z}_K$, is a weighted voting regression denoted by $F_T(\mathbf{z}_d)$ based on $T$ weak local experts, which are regression trees in our work, each is denoted by $f_t(\mathbf{z}_d) \in \mathbb{R}$, $t = 1, \ldots, T$. The proposed F1ERCE algorithm then collaboratively trains the local experts using the actual channel vectors $h_d, d = 1, ..., D$ in the training set corresponding to each measurement $\mathbf{z}_d$. The algorithm is detailed in Algorithm 1 and the final estimator $F_T$ F1ERCE is seeking for all measurements can be described as follows:

$$\begin{bmatrix} \hat{\mathbf{h}}_1 \\ \hat{\mathbf{h}}_2 \\ \vdots \\ \hat{\mathbf{h}}_d \end{bmatrix} = F_T \left( \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_d \end{bmatrix} \right)$$

**Algorithm 1:** F1ERCE

**Result:** Channel Estimation

**Initialization**;

Uncorrelate and compress the input data matrix $\mathbf{Z}$ using PCA to $K$ dimensions $\mathbf{Z}_K$;

Set the learning rate $\eta \in ]0, 1]$;

Set the initial regression additive function to the average channel value $F_0(\mathbf{z}_k) = \frac{1}{D} \sum_d \mathbf{h}_d$;

**for** $t \leftarrow 1$ **to** $T$ **do**

  **for** $d \leftarrow 1$ **to** $D$ **do**

    Compute the residual error between the estimated and actual channels for the $d$th sample according to $\mathbf{r}_d = \mathbf{h}_d - F_{t-1}(\mathbf{z}_d)$;

    Fit the local expert $f_t(\mathbf{z}_d)$ by using tree regression to $\mathbf{r}_d$;

  **end**

  Update $F_t(\mathbf{z}_d) = F_{t-1}(\mathbf{z}_d) + \eta * f_t(\mathbf{z}_d)$;

**end**

**Final Regression** $\hat{\mathbf{h}_d} = F_T(\mathbf{z}_d), d = 1, ..., D$

In the next section, we present our numerical results, which prove the efficacy of the proposed algorithm.

## IV. Simulation Results

The simulation results are generated using a modified version of the code from [8]. The network parameters used are those described in [8] to allow for proper comparison of performance. The network layout is a conference room of dimensions $10 \times 10 \times 5$ meters with two tables. The BS antennas are 2.5 m high, and the user is simulated in 150k locations in the room at 1 m height.

The proposed F1ERCE in massive MIMO has several hyper-parameters to tune. First, the number of PCA components, which is chosen to be 200, a figure which retains more than 95% of the information for different numbers of antennas. Next, the number of trees and the properties of each tree. There is always a trade-off between bias (generalization, which can result in under-fitting) and variance (complexity, which can result in over-fitting). In general, decreasing the trees depth and increasing their number produces better generalization [15]. Yet, of course increasing the number of trees demand more computations. The balance reached with cross-validation is to have 500 trees and 5 splits maximum per tree. Finally, the learning rate is chosen to be 0.9, which is based on empirical rules of thumb [13].

TABLE I
SUMMARY OF SIMULATION AND MODEL PARAMETERS

| | |
|---|---|
| Network Layout Parameters | |
| Room Size | $10 \times 10 \times 5$ m |
| User height | 1 m |
| Active BSs | 32 |
| Bandwidth | 0.01 GHz |
| Number of Antennas in y-axis | from $2 - 100$ |
| Number of multipaths | 1 |
| ER Parameters | |
| Number of weak trees | 500 |
| Number of splits per tree | 5 |
| Learning rate | 0.9 |
| Boosting algorithm | LSBoost |

Table I summarizes the network layout simulation and the ER learning parameters.

### A. The proposed model performance for different pilot lengths

Fig. 3 shows the results of the proposed model and the DL model proposed by [8] for different pilot lengths and various number of antennas. It can be readily seen that the general trend of performance behavior follows that of [8]; the more antennas, the more paths seen by the model, and the better the overall performance is. Moreover, the F1ERCE algorithm is able to perform better when the number of antennas is intermediate. This is a result of the adaptive capacity the ER uses to avoid over-fitting through its step-by-step growing of complexity that is guided by modelling residual errors as shown in Algorithm 1.

The capacity of the utilized ER model with 500 regression trees starts to be under-fit when the number of antennas reaches 50. To overcome this, simply more tress should be added. Fig. 4 shows how increasing the number of regression tress improves the F1ERCE capacity to learn and improve performance at pilot length 5 and 50 antennas.

For the utilized model, the savings in computations are significant. The previous attempt to solve the channel estimation problem using highly quantized inputs utilized a huge fully-connected network to achieve the task [8]. For example, in the case for pilot length of 5, the neural network model requires approximately 72M multiplication operations and comparisons (because the non-linearity used is a rectified-linear unit) to do an estimation for just one observation. On the other hand, our proposed model requires approximately 50k divisions for voting, 500 multiplications for PCA, and 250k comparisons for estimations in the regression trees (relational operations).

Moreover, according to [16], the average CPU power required for different clock frequencies is shown in Table II. The total power consumed given the required amount of computations for a single observation in the case of pilot length 5 would then be approximately 21 J for the DL model, and 0.8 J for F1ERCE.

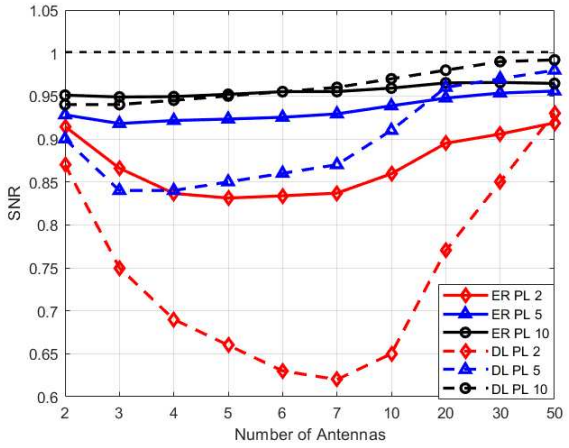


Fig. 3. SNR per antenna for different pilot lengths and number of antennas.

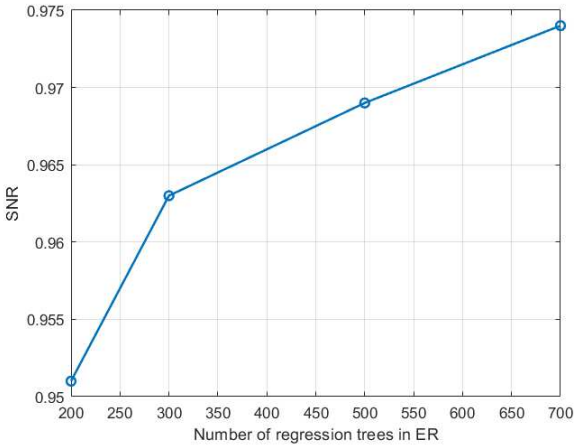


Fig. 4. SNR per antenna vs. the number of regression trees with pilot length 5 and 50 antennas.

That accounts for 96% computational power requirements reduction on average.

### B. The proposed model performance for different training sizes

The number of samples required for training was estimated properly in [8]. However, in real life scenarios, such number of

TABLE II
AVERAGE POWER CONSUMPTION FOR ARITHMETIC OPERATIONS

| Operation | Average Power Consumption |
|---|---|
| Comparison | 194.5 nJ |
| Multiplication | 296.5 nJ |
| Division | 325.5 nJ |

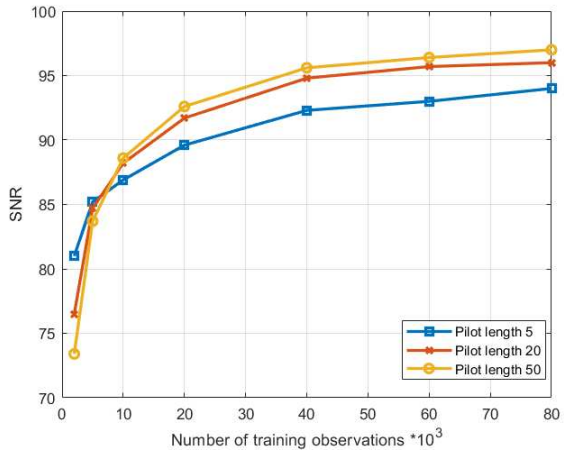


Fig. 5. SNR per antenna for different pilot lengths and number of antennas.

training observations may not be available to adapt the system when there is a change in the nature of the channel. Fig. 5 shows the impact of training the proposed model using different number of observations versus the performance for a pilot length of 5 and different number of antennas. It is clear that the rate of deterioration with decreasing the number of training samples is correlated to the size of input dimensions. Again, this figure assesses the ER resilience against over-fitting when the data decreases, and it shows that the deterioration due to lack of data is sustainable at 5% even after dropping 80% of the total amount of training data utilized in [8]. Of course, this will not be the case for DL, which has much more parameters to learn and depends on back propagation of total error rather than adaptive step-by-step learning that is based on residual errors.

## V. Conclusion

This work proposed a practical ER solution for channel estimation based on 1-bit quantized observations that is computationally and power efficient. The proposed ER algorithm uses regression trees as its building unit, which can be easily implemented as combinational logic circuits. The proposed model achieves over 97% reduction in the number of required computations required by the previously proposed model in the literature and 96% reduction in the power requirements while sustaining similar performance in most cases. The work also shows the impact of the training size on the performance of the proposed model. It is found that the proposed algorithm is more robust against data scarcity for training and adaptation. In conclusion, the proposed ER model shows high potential for being a practical solution for the 1-bit quantization-based channel estimation problem.

## Acknowledgements

The work of Ahmed Elsheikh and Mahmoud H. Ismail is supported by the Smart Cities Research Institute (SCRI) at the American University of Sharjah under grant EN0281:SCRI18-07. The work of Ahmed S. Ibrahim is supported in part by the National Science Foundation under Award No. CNS-2144297.

## References

[1] J. Mo, P. Schniter, and R. W. Heath, "Channel estimation in broadband millimeter wave MIMO systems with few-bit ADCs," *IEEE Transactions on Signal Processing*, vol. 66, no. 5, pp. 1141–1154, 2017.

[2] B. Murmann, "Energy limits in A/D converters," *2013 IEEE Faible Tension Faible Consommation*, pp. 1–4, 2013.

[3] C.-K. Wen, C.-J. Wang, S. Jin, K.-K. Wong, and P. Ting, "Bayes-optimal joint channel-and-data estimation for massive MIMO with low-precision ADCs," *IEEE Transactions on Signal Processing*, vol. 64, no. 10, pp. 2541–2556, 2015.

[4] J. Choi, J. Mo, and R. W. Heath, "Near maximum-likelihood detector and channel estimator for uplink multiuser massive MIMO systems with one-bit ADCs," *IEEE Transactions on Communications*, vol. 64, no. 5, pp. 2005–2018, 2016.

[5] S. Jacobsson, G. Durisi, M. Coldrey, U. Gustavsson, and C. Studer, "One-bit massive MIMO: Channel estimation and high-order modulations," in *2015 IEEE International Conference on Communication Workshop (ICCW)*, pp. 1304–1309, IEEE, 2015.

[6] T.-C. Zhang, C.-K. Wen, S. Jin, and T. Jiang, "Mixed-ADC massive MIMO detectors: Performance analysis and design optimization," *IEEE transactions on wireless communications*, vol. 15, no. 11, pp. 7738–7752, 2016.

[7] H. Wang, T. Liu, C.-K. Wen, and S. Jin, "Optimal data detection for OFDM system with low-resolution quantization," in *2016 IEEE International Conference on Communication Systems (ICCS)*, pp. 1–6, IEEE, 2016.

[8] Y. Zhang, M. Alrabeiah, and A. Alkhateeb, "Deep learning for massive mimo with 1-bit adcs: When more antennas need fewer pilots," *IEEE Wireless Communications Letters*, vol. 9, no. 8, pp. 1273–1277, 2020.

[9] S. Gao, P. Dong, Z. Pan, and G. Y. Li, "Deep learning based channel estimation for massive MIMO with mixed-resolution ADCs," *IEEE Communications Letters*, vol. 23, no. 11, pp. 1989–1993, 2019.

[10] E. Balevi and J. G. Andrews, "One-bit OFDM receivers via deep learning," *IEEE Transactions on Communications*, vol. 67, no. 6, pp. 4326–4336, 2019.

[11] A. Klautau, N. González-Prelcic, A. Mezghani, and R. W. Heath, "Detection and channel equalization with deep learning for low resolution MIMO systems," in *2018 52nd Asilomar Conference on Signals, Systems, and Computers*, pp. 1836–1840, IEEE, 2018.

[12] Y.-S. Jeon, S.-N. Hong, and N. Lee, "Supervised-learning-aided communication framework for MIMO systems with low-resolution ADCs," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 8, pp. 7299–7313, 2018.

[13] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[14] A. M. Abdelsalam, A. Elsheikh, J.-P. David, and J. P. Langlois, "POLYBiNN: a scalable and efficient combinatorial inference engine for neural networks on FPGA," in *2018 Conference on Design and Architectures for Signal and Image Processing (DASIP)*, pp. 19–24, IEEE, 2018.

[15] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[16] A. Leite, C. Tadonki, C. Eisenbeis, and A. de Melo, "A fine-grained approach for power consumption analysis and prediction," *Procedia Computer Science*, vol. 29, pp. 2260–2271, 2014.