

Smart metering system data analytics platform using multicore edge computing

Juan C. Olivares-Rojas¹, Enrique Reyes-Archundia², José A. Gutiérrez-Gnecchi³, Ismael Molina-Moreno⁴, Adriana C. Téllez-Anguiano⁵, Jaime Cerda-Jacobo⁶

^{1,2,3,4,5}División de Estudios de Posgrado e Investigación, Tecnológico Nacional de México / I.T. de Morelia, México

⁶Facultad de Ingeniería Eléctrica, Universidad Michoacana de San Nicolás de Hidalgo, México

Article Info

Article history:

Received Dec 2, 2020

Revised Jan 10, 2021

Accepted Jan 20, 2021

Keywords:

Data analytic

Edge computing

GPU

Internet of things

Smart meter

ABSTRACT

The smart grid revolution has only been possible, thanks to the development and proliferation of smart meters. The increasingly growing computing capabilities for Internet of Things devices have made it possible for data to be processed directly from the devices where it is produced; this has been called edge computing. Edge computing is allowing the smart grid to become increasingly intelligent to solve problems that make electricity consumption more efficient and environmentally friendly. This work presents the implementation of a smart metering system that allows data analytics using a multiprocessing architecture directly on the smart meter. The results show that the development of smart meters with data analytics capabilities at the edge is a reality today, and the use of multiprocessing permits the improvement of data processing.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Juan C. Olivares-Rojas

División de Estudios de Posgrado e Investigación

Tecnológico Nacional de México / Instituto Tecnológico de Morelia

Av. Tecnológico 1500, Lomas de Santiaguillo, Morelia, Michoacán 58129, México

Email: juan.or@morelia.tecnm.mx

1. INTRODUCTION

The world has been revolutionized thanks to the technologies that are emerging within the so-called fourth industrial revolution. This revolution is characterized by the rise of cyber-physical systems (CPS) whose fundamental characteristic is the integration of physical objects with people, through the technologies and the data which is being generated [1]. The CPS are transforming the electrical industry allowing to have a more reliable, cleaner and cost-effective power grid. The electricity sector is of utmost importance for the development of the industry, economy, and society in general, which is why the improvement of the grid affects practically all human activities. The inclusion of CPS has provided the grid with data analytics and communication capabilities, which provides it with certain intelligence, which is why it has been called the smart grid (SG) [2].

The SG started with the inclusion of smart metering systems (SMS) whose main component is smart meters (SM). SMS allows to know the electricity consumption of end users through web portals or mobile applications that permit the visualization of information such as consumption patterns and the cost of said energy. SMs can be viewed as Internet of Things (IoT) devices because they are embedded systems with wired and wireless connectivity capabilities. SMS also allows you to handle electrical system operations such as outages and reconnections automatically. In recent years, SMs allow measuring the electricity production of end users called prosumers, which through distributed energy resources (DER) can produce electrical energy that helps them lower their energy costs [3].

One of the main trends in distributed computing is to divide computing and processing capabilities into three levels, called: cloud, fog, and edge. Edge computing refers to running processes closer to where the sensor and actuator data is acquired in the end devices. End-end devices are generally IoT devices or embedded systems with more limited processing and storage capabilities than devices in the fog and the cloud [4]. Although in recent years, the capacity of hardware systems for IoT has increased, making them increasingly available for data analytics and machine learning applications, even in many IoT applications, more significant hardware features are required.

Currently, SMs are beginning to have greater computing capacity that makes them more viable to carry out data analytics and machine learning processes directly on the device, which is why edge computing is beginning to be applied in SMs. Significant computing power is required for some data analytics and machine learning applications in SM. Traditionally, multi-core processing systems, multiprocessors, or acceleration hardware such as graphics processing units (GPUs) and computational neural sticks, among others, have been used [5].

Many simple board computers (SBC) are using multicore processors and even GPUs, such as the case of Google Coral AI and Nvidia Jetson Nano, to name a few. Although the most current operating systems and compilers can support the development of concurrent and parallel processing systems, the design of algorithms that can be executed concurrently and in parallel is still required. The present work shows an SM that uses parallel edge computing in SMS for data analytics applications. The results show a significant improvement compared to the use of SM with traditional processing. This paper is structured as follows. Section 2. presents a short review of the literature regarding data processing in parallel in edge computing, while Section 3. shows the architecture proposed for the development of edge computing in SMs. The results obtained are discussed in Section 4. Finally, in Section 5. the conclusions and future work in this research area are presented.

2. PARALLEL PROCESING IN EDGE COMPUTING

Most of the multiprocessing related works in SG have focused on the cloud or fog part: in both cases, there are sufficient computing capacities to perform data analytics and machine learning processes as in [5-7]. Recently much of the multiprocessing work in edge computing has focused on image processing and computer vision taking advantage of the capabilities of the GPUs that are being incorporated into SBCs: an example of these are the [8-12] related works. The use of SBC with the integration of GPU use has been studied by various works [13-15] in which the Jetson Nano board stands out for its excellent performance and low costs. On the other hand, computing processing is being carried out on edge with this type of board for different contexts, such as sound management and error detection [16, 17].

One of the main characteristics that a multiprocessing system should have is the handling of appropriate algorithms to be executed in this class of concurrent and parallel hardware platforms: for this reason, various authors have focused on the parallelization of forecasting and classifying algorithms that allow improvements in the results of process time and effectiveness [18-21]. In the past year, there was an exponentially growing interest in edge computing applications using SBC like Jetson Nano. For instance, in [22] a novel erasure-code system is presented. In [23] the authors present deep learning techniques optimized for edge computing. Other works are focused on enhanced biometrics security in edge devices [24]. In [25] the authors present a Recurrent Neural Network (RNN) for classification power systems contingences using edge devices. Other novel diverse applications using Jetson Nano for edge computing are presented in works [26-29]. In general, edge computing applications for data analytics in the smart grid are only just beginning to emerge, and as IoT hardware becomes more capable of data analytics and machine learning applications, works in the field of SMS will appear.

3. FORECASTING IN SMART METERING SYSTEM USING A MULTICORE ARCHITECTURE

A SM was developed using a Jetson Nano board using current and voltage sensors. Figure 1 shows a diagram of the hardware architecture used for the acquisition of electrical signal values using a SmartPi energy board the connection between the two boards is made through the General Input and Output Ports (GPIO). SmartPi essentially works with the Raspberry Pi but can be adapted to the Jetson Nano with some modifications. Household appliances and Distributed Energy Resources (DER) such as photovoltaic panels and wind turbines are connected through each of the three phases of the three-phase meter to its corresponding voltage and current sensor.

The data values used are current intensity (A), voltage (V), active power (W), power factor, reactive power (W), energy consumed (kWh), and energy produced (kWh). One of the main problems in the SG is

forecasting the energy consumption/production of the prosumers. For this reason, we choose the implementation of a forecasting algorithm for testing the SM architecture using the time series approach. Due to the use of large time series is very complex for IoT embedded devices, it is necessary to improve the forecasting model's performance at edge computing.

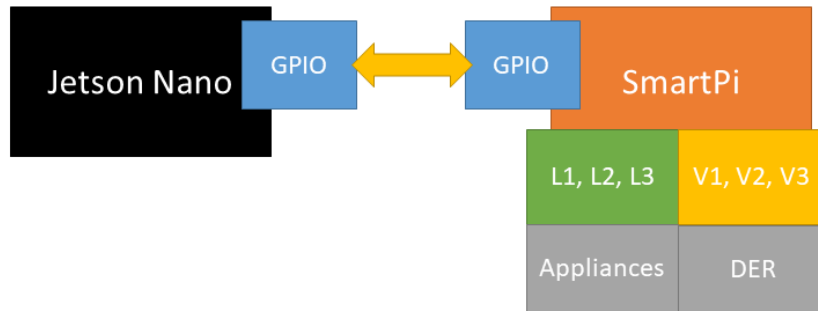


Figure 1. Hardware architecture of the multicore smart meter

Jetson nano is an SBC with a Quad-core ARM microprocessor with a GPU included with 128-cores. The cores were used to develop a parallel power consumption forecast algorithm to detect possible anomalies that could be considered failures. The algorithm used is described in Algorithm 1.

Algorithm 1. Parallel ARIMA

Input: A Time Series (TS) with energy values, the Size (S) of the partition, the Time (T), the initial p, d, q of the ARIMA model.

- 1: Split the data set of size S according to the indicated time T
- 2: Assign each core an ARIMA (p_i, d_i, q_i) process
- 3: Compute the ARIMA (p_i, d_i, q_i) values of the TS that do not overlap
- 4: For all series whose values overlap, integrate the missing values with the data set S_{i+1}
- 5: Integrate all results

Output: comparison of the TS data with itself using metric like Relative Absolute Error (RAE)

In general, the algorithm uses the ARIMA technique dividing the data by periods depending on the granularity of the study time, which can be per day, week, month, or year. With data segmentation, the calculations are processed in parallel according to the indicated time window, except for the data that intersects between periods that need to be completed separately. Once the computations have been carried out in parallel, the results are gathered. Algorithm 1. was implemented using the Python language through the statsmodels library using the PyCUDA framework for handling parallel calculations through the various GPU cores. The implementation tries to use the 128 available cores.

4. RESULTS AND DISCUSSION

To compare, a traditional sequential classification algorithm was used, comparing it with the parallel algorithm described previously in Algorithm 1. Since there are few measurement data (14 months of readings), it was decided to use the data sets described in IEEE DataPort [30] and UCI Machine Learning [31] with their adaptation of data to the variables of interest in this work (Timestamp and Energy Consumption) grouping according the period of evaluation. The IEEE Dataset has a lot of variables and it has data for 3248 SM in one year. The readings are in slots of 30 minutes. The data must be grouped according of each SM.

The UCI dataset has information of only one meter in slots of 15-minutes. The energy has expressed in each of the three phases; for this reason, the energy consumption must be summarized. The total number of instances are 2,075,259. The readings are from between December 2006 and November 2010 (47 months). Table 1 shows the results of the execution of forecasting models using linear architecture comparing them with their parallel equivalent, taking into account different partition sizes: week, month, day, and year for the IEEE dataset, while in Table 2 shows the same results but for the UCI dataset. The times are expressed in minutes. Both models have the same accuracy.

The results show a considerable decrease in execution times in both cases, although they are dependent on the size of the data and the granularity of the information analysis. The greater the amount of data, the longer the processing time. The complexity of ARIMA sequential is equivalent to $O(n)$ and the parallel version is equivalent to $O(n \log n) + C$ where C is the delay time to integrate the non-parallel data. Both models (parallel and non-parallel) have the same accuracy $R^2=89.97\%$ with the IEEE dataset and $R^2=83.25\%$ with the UCI dataset.

Table 1. Parallel ARIMA with IEEE dataset

Model	Processing Times			
	Day	Week	Month	Year
Sequential	147.76	43.64	4.99	1.33
Parallel	9.27	5.47	3.06	1.02

Table 2. Parallel ARIMA with UCI dataset

Model	Processing Times			
	Day	Week	Month	Year
Sequential	248.53	57.25	6.38	1.72
Parallel	12.39	6.75	3.80	1.23

There are other methods/techniques for anomaly detection that could be implemented in this parallel framework. Still, not all methods/techniques are suitable to be programmed in parallel due to restrictions of any parallel algorithm. Particularly, the simplest implementations are better for IoT and embedded devices.

5. CONCLUSION AND FUTURE WORK

The use of edge computing is growing day by day, allowing IoT devices to have greater functionality. In the case of the smart grid, many IoT devices are benefiting from the management of computing at the edge, and particularly in SMS, it will be widely used to improve data analytics processes through various statistical, probabilistic, and machine learning and artificial intelligence. The integration of multiprocessing architectures in the IoT hardware will make it possible to improve execution times in forecasting and data classification models in edge computing. For this, it will be necessary to adapt the algorithms so that they can be executed in parallel. This work demonstrates that it is already feasible to perform edge computing using a multiprocessing architecture in smart metering systems. The results show a considerable reduction in processing times that make it a viable option for processing data closer to the source of the data.

As future work in the first instance, the design of a more elaborate data processing algorithm is contemplated to take advantage of the computing capabilities of SM. Besides, it is contemplated the integration of this architecture of measurement and data analytics using computing schemes at the level through data concentrators and computing in the cloud using the servers of the Metering Database Management Systems (MDMS).

ACKNOWLEDGEMENTS

This work is partially supported by Tecnológico Nacional de México under grants 7948.20-P and 8000.20-P.

REFERENCES

- [1] S. Bonilla, *et al.*, "Industry 4.0 and sustainability implications: A scenario-based analysis of the impacts and challenges," *Sustainability*, vol. 10, no. 10, 2018, doi: 10.3390/su10103740.
- [2] G. Konstantopoulos, *et al.*, "Towards the integration of modern power systems into a cyber-physical framework," *Energies*, vol. 13, no. 9, p. 2169, 2020, doi: 10.3390/en13092169.
- [3] M. Daneshvar and S. Asadi, "CPS-based transactive energy technology for smart grids," in *Cyber-Physical Systems in the Built Environment*, Springer, 2020, pp. 323-338, doi: 10.1007/978-3-030-41560-0_18.
- [4] M. Forcan and M. Maksimović, "Cloud-Fog-based approach for smart grid monitoring," *Simulation Modelling Practice and Theory*, vol. 101, pp. 1-18, 2020, doi: 10.1016/j.simpat.2019.101988.

- [5] Z. Li and F. Yang, "Advanced metering infrastructure and graphics processing unit technologies in electric distribution networks," in *Electric Distribution Network Management and Control Power Systems*, Springer, 2018, pp. 309-345, doi: 10.1007/978-981-10-7001-3_12.
- [6] S. Grubmüller, et al., "Predictive energy management on multi-core systems," in *Comprehensive Energy Management - Safe Adaptation, Predictive Control and Thermal Management*, Springer, 2018, pp. 47-66, doi: 10.1007/978-3-319-57445-5_4.
- [7] O. Debauche, S. Mahmoudi, S. A. Mahmoudi, P. Manneback, J. Bindelle, and F. Lebeau, "Edge computing and artificial intelligence for real-time poultry monitoring," *Procedia Computer Science*, vol. 175, pp. 534-541, 2020, doi: 10.1016/j.procs.2020.07.076.
- [8] P. Álvarez and M. Luís, "Configuración y ejecución de algoritmos de visión artificial en la tarjeta Nvidia Jetson TK1 DevKit," 2017, [Online]. Available: <https://ebuah.uah.es/dspace/handle/10017/30204>.
- [9] A. Basulto-Lantsova, J. A. Padilla-Medina, F. J. Perez-Pinal, and A. I. Barranco-Gutierrez "Performance comparative of OpenCV Template Matching method on Jetson TX2 and Jetson Nano developer kits," in *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, USA, 2020, pp. 0812-0816, doi: 10.1109/CCWC47524.2020.9031166.
- [10] P. Inthanon and S. Mungsing, "Detection of drowsiness from facial images in real-time video media using nvidia Jetson Nano," *2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, Phuket, Thailand, 2020, pp. 246-249, doi: 10.1109/ECTI-CON49241.2020.9158235.
- [11] W. Vajitkunsawat and P. Chantngarm, "comparison of machine learning algorithm's on self-driving car navigation using Nvidia Jetson Nano," on *2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, Phuket, Thailand, 2020, pp. 201-204, doi: 10.1109/ECTI-CON49241.2020.9158311.
- [12] S. Ullah and D. Kim, "Benchmarking Jetson platform for 3D point-cloud and hyper-spectral image classification," in *2020 IEEE International Conference on Big Data and Smart Computing (BigComp)*, Busan, Korea (South), 2020, pp. 477-482, doi: 10.1109/BigComp48618.2020.00-21.
- [13] S. Cass, "Nvidia makes it easy to embed AI: The Jetson nano packs a lot of machine-learning power into DIY projects," *IEEE Spectrum*, vol. 57, no. 7, pp. 14-16, 2020, doi: 10.1109/MSPEC.2020.9126102.
- [14] V. Mazzia, A. Khaliq, F. Salvetti, and M. Chiaberge., "Real-time apple detection system using embedded systems with hardware accelerators: An edge AI application," *IEEE Access*, vol. 8, pp. 9102-9114, Jan. 2020, doi: 10.1109/ACCESS.2020.2964608.
- [15] A. Süzen, et al., "Benchmark analysis of Jetson TX2, Jetson Nano and Raspberry PI using Deep-CNN," in *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, Ankara, Turkey, 2020, pp. 1-5, doi: 10.1109/HORA49412.2020.9152915.
- [16] C. Aguilera, et al., "Fast CNN stereo depth estimation through embedded GPU devices," *Sensors*, vol. 20, no. 11, p. 3249, 2020, doi: 10.3390/s20113249.
- [17] J. Seo, Hanse Ahn, Daewon Kim, Sungju Lee, Yongwha Chung, and Daihee Park, "EmbeddedPigDet—Fast and accurate pig detection for embedded board implementations," *Applied Sciences*, vol. 10, no. 8, p. 2878, 2020, doi: 10.3390/app10082878.
- [18] M. Khashei and Z. Hajirahimi, "Performance evaluation of series and parallel strategies for financial time series forecasting," *Financial Innovation* vol. 3, no. 1, pp. 1-24, 2017, doi: 10.1186/s40854-017-0074-9.
- [19] Stokely, M., et al., "Large-scale parallel statistical forecasting computations," in *R. JSM Proceedings, Section on Physical and Engineering Sciences, American Statistical Association*, 2011. [Online]. Available: <https://research.google/pubs/pub37483/>.
- [20] H. R. Rangel, G. D. Rocca, J. J. Flores, L. A. M. Rosales, and N. E. C. Garcia, "Implementación de kNN sobre un GPU para predicción de la velocidad del viento," *Research in Computing Science*, vol. 147, no. 5, pp. 239-253, 2018, doi: 10.13053/rcs-147-5-18.
- [21] J. Trapero, J., "Implementation of probabilistic forecasts in a GPU for big data problems," 2019, [Online]. Available: <https://www.semanticscholar.org/paper/Implementation-of-probabilistic-forecasts-in-a-GPU-Trapero/a4c7c8733883bb5e943c8ba4dc5155e61d0321c1>.
- [22] L. Liang, H. He, J. Zhao, C. Liu, Q. Luo and X. Chu, "An erasure-coded storage system for edge computing," in *IEEE Access*, vol. 8, pp. 96271-96283, 2020, doi: 10.1109/ACCESS.2020.2995973.
- [23] G. Li, X. Ma, X. Wang, L. Liu, J. Xue and X. Feng, "Fusion-catalyzed pruning for optimizing deep learning on intelligent edge devices," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 39, no. 11, pp. 3614-3626, Nov. 2020, doi: 10.1109/TCAD.2020.3013050.
- [24] K. Saeed, M. Adamski, A. Klimowicz, A. Lupinska-Dubicka, M. Omieljanowicz, G. Rubin, M. Rybnik, M. Szymkowski, M. Tabedzki, and L. Zienkiewicz, "A novel extension for e-Safety initiative based on developed fusion of biometric traits," in *IEEE Access*, vol. 8, pp. 149887-149898, 2020, doi: 10.1109/ACCESS.2020.3015503.
- [25] W. Hong, D. Huang, C. Chen and J. Lee, "Towards accurate and efficient classification of power system contingencies and cyber-attacks using recurrent neural networks," in *IEEE Access*, vol. 8, pp. 123297-123309, 2020, doi: 10.1109/ACCESS.2020.3007609.
- [26] I. A. Lungu, A. Aimar, Y. Hu, T. Delbruck and S. -C. Liu, "Siamese networks for few-shot learning on edge embedded devices," in *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 10, no. 4, pp. 488-497, Dec. 2020, doi: 10.1109/JETCAS.2020.3033155.

- [27] C. Gao, A. Rios-Navarro, X. Chen, S. -C. Liu and T. Delbruck, "EdgeDRNN: Recurrent neural network accelerator for edge inference," in *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 10, no. 4, pp. 419-432, Dec. 2020, doi: 10.1109/JETCAS.2020.3040300.
- [28] J. Fontaine, M. Ridolfi, B. V. Herbruggen, A. Shahid and E. De Poorter, "Edge inference for UWB ranging error Correction Using Autoencoders," in *IEEE Access*, vol. 8, pp. 139143-139155, 2020, doi: 10.1109/ACCESS.2020.3012822.
- [29] J. Lin, C. Gan, K. Wang and S. Han, "TSM: Temporal shift module for efficient and scalable video understanding on edge devices," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020, doi: 10.1109/TPAMI.2020.3029799.
- [30] Triguero, "IEEE-CIS technical challenge on energy prediction from smart meter data," *IEEE DataPort*, 2020, doi: <https://dx.doi.org/10.21227/2npg-c280>.
- [31] G. Hébrail. *et al.*, "Individual household electric power consumption Data Set," *UCI Machine Learning*, 2012. [Online]. Available: <https://archive.ics.uci.edu/ml/datasets/individual+household+electric+power+consumption>.

BIOGRAPHIES OF AUTHORS



Juan C. Olivares-Rojas received his BEng degree in Computer Systems at Instituto Tecnológico de Morelia and his MSc degree in computer sciences at National Center for Research and Technological Development (CENIDET). He is pursuing a Ph.D. in engineering sciences from Instituto Tecnológico de Morelia. He is full-time professor at Department of systems and computing, Instituto Tecnológico de Morelia. His current research interests include cyber security, smart grid and distributed systems.



Enrique Reyes-Archundia received his BEng degree in Electronics and Ph.D. in Electrical Engineering at Instituto Tecnológico de Morelia and his MSc degree in Electronics at National Center for Research and Technological Development (CENIDET). He is full-time professor at electronics graduated program, Instituto Tecnológico de Morelia. His current research interests include control systems for smart grid and signal processing.



José A. Gutiérrez-Gnecchi received his BEng degree in Industrial Electronics at Instituto Tecnológico de San Luis Potosi, and his MSc degree in Instrumentation and Analytical Science and his Ph.D. in Electrical and Electronics Engineering at the University of Manchester. He is full-time professor at electronics graduated program, Instituto Tecnológico de Morelia. His current research interests include Smart Metering Systems in diverse fields such as biomedical and agricultural.



Ismael Molina-Moreno received his BEng degree in Industrial Electronics at Instituto Tecnológico de San Luis Potosi, his MSc degree in Electronics at Technological Institute of Morelia, and his Ph.D. at Universidad Michoacana. He is full-time professor at electronics graduated program, Instituto Tecnológico de Morelia. His current research interests include power quality state estimation, powers electronics, and embedded systems.



Adriana Téllez-Anguiano received his BRng degree in Electronics at Instituto Tecnológico de Morelia, and her MSc. in Electronics and Ph.D. at Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET). She is full-time professor at electronics graduated program, Instituto Tecnológico de Morelia. Her research lines are Signal Processing, Automatic Control and Renewables Energies.



Jaime Cerda-Jacobo received the Electrical Engineer degree from Universidad Michoacana de San Nicolas de Hidalgo, Morelia, Mexico, in 1990, the M.Sc. in computer science from the Research and Advanced Studies Center at IPN, Mexico City, Mexico in 2000 and the PhD degree in electrical and electronic engineering from University of Southampton, UK in 2010. He is with the Universidad Michoacana as Associate Professor. His actual research activities concentrate in the areas of machine learning, optimization and its decentralization by graph manipulations.