Edge data processing using Kalman filter

1st Gabriel TUDOR
Department of Power Engineering Systems
Polytechnic University of Bucharest – Faculty of Power Engineering
Bucharest, Romania
gabi.tudor912@gmail.com

2nd Cornel-Cristian ANDREI
Department of Power Engineering Systems
Polytechnic University of Bucharest – Faculty of Power Engineering
Bucharest, Romania
cornel.andrei@stud.energ.upb.ro

3rd Madalina ARHIP-CALIN
Department of Power Engineering Systems
Polytechnic University of Bucharest – Faculty of Power Engineering
Bucharest, Romania
madalina.arhip@upb.ro

4th Valentina ROHAT
Department of Power Engineering Systems
Polytechnic University of Bucharest – Faculty of Power Engineering
Bucharest, Romania
valentinarohat@yahoo.com

2021 9th International Conference on Modern Power Systems (MPS) | 978-1-6654-3382-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/MPS52805.2021.9492575

Abstract— **The evolution of the modern power grid is closely tied to the advances in the telecommunications field. Data transmission from the field no longer poses a problem, but the large volume of data generated that needs to be handled, continues to be a challenge. As a direct result of the continuous push for digital technology integrations within the power grid, Electric Utilities have gained extended capabilities with the help of SCADA (Supervisory Control And Data Acquisition) system and its various applications used for control and visualization of real-time data received from the process. SCADA systems help human operators to make informed decisions and integrates various devices in order to achieve Smart Grid capabilities by using a Distributed Control System architecture to control geographically dispersed assets. Smart Grids utilize a series on services and products in order to promote the penetration of renewable resources, together with control, monitoring and self-healing technologies. This paper aims to explore the role that Edge plays in the flow of data, starting from the place where data is generated until it reaches the Cloud.**

Keywords—Smart Grid. Edge, Fog, Cloud, IOT, I-IOT

I. INTRODUCTION

Digital communication in a Smart Grid includes a series of protocols and devices, this includes IoT (Internet of Things) and I-IoT (Industrial Internet of Things) devices. IoT defines itself as a vast digital network of "things" which are capable of transmitting data over the network to the Cloud for further processing.

I-IoT represents and extension of the conventional IoT, developed with the intention to solve industry specific constraints:

- The need for real time data analysis – As opposed to conventional IPv4 networks data coming from IoT devices require real time processing. Relocating the analysis software closer to the Edge where data is produced and assuring real time data ingestion combined with relational databases at the IP level is a necessity.
- Legacy support – An IoT network is made up of conventional IPv4 networks and legacy systems with non-IP devices and proprietary communication protocols over serial communication. IoT networks need to support protocol conversion and offer tunneling mechanisms to connect legacy systems to standard IP and Ethernet protocols.
- Large volume of data – A large network of sensors will generate a large amount of data which leads to bandwidth limitations and slower transfer rates. To reduce the transfer of large volumes of data over vast distances, computing resources need to be distributed from Edge to Cloud.
- IoT device constraints – Limitations regarding processing resources, power (devices that are battery powered) and memory (volatile and non-volatile).
- Conventional IPv4 addressing is not feasible due to exhaustion of assignable addresses. Scalability is huge and requires more scalable routing protocols. Transitioning from IPv4 addressing towards IPv6 is necessary in order to assure network connectivity and it is done using an Adaptation Layer (Fig.1).

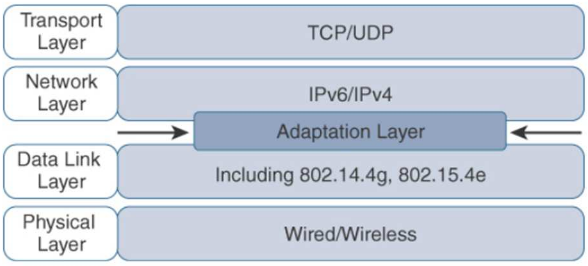


Fig. 1. Adaptation Layer[3]

Security – Exposed IoT devices. Device level authentication is required and the encryption of network traffic.

Communication inside IP based networks is typically described by the OSI model (Open System Interconnection) developed by the International Organization for Standardization (ISO) and from a conceptual standpoint, offers a series of advantages:

- It simplifies communication by dividing network entities into seven subsystems (layers) which eases network debugging and development.
- Introduces standardization, encouraging the development of network devices by multiple vendors.
- Offers a clear definition of the functions that each layer of the stack must ensure.
- Increased connection between software and hardware of different types.
- Allows upgrades to any layer of the stack.

978-1-6654-3382-2/21/$31.00 ©2021 IEEE

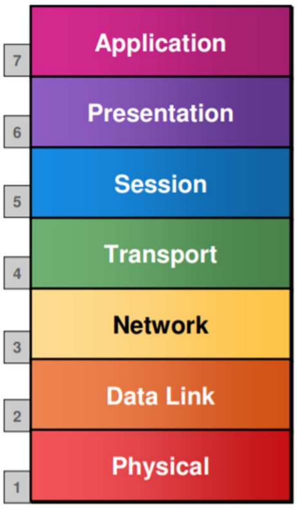


Fig. 2. OSI stack [6]

Layer 1 (Physical Layer) connects end devices to the upper layers of the OSI stack, and represents the medium through which raw data (Bits) is transmitted. This layer describes the physical properties of cables, connectors (number of pins and voltage levels). Standards associated with this layer:

- IEEE 802.3 (Ethernet)
- IEEE 802.11(Wireless)

Layer 2 (Data-Link) takes the raw data from Layer 1 and organizes it into frames at the Switch level. Access is done by MAC (Media Access Control) addressing, unique for every device. This layer is also responsible for error detection at the Physical Layer.

- IEEE 802 (software and hardware connection)

Layer 3 (Network) organizes data into packets at the Router level. The Router (also referred as gateway) is responsible for connecting different networks and uses routing protocols to find the shortest path to transmit packets.

Layer 4 (Transport) organizes data into segments and is responsible for error correction. The rules for communication are established using TCP (Transmission Control Protocol) and UDP (User Datagram Protocol).

Layer 5 (Session) controls the dialog between computers by setting up and tearing down sessions.

Layer 6 (Presentation) formats data in a manner that is recognizable by the Application Layer. Offers encryption, decryption and compression.

Layer 7 (Application) services developed for user interaction at the Enterprise level.

II. SCADA SYSTEM OVERVIEW

Supervisory Control And Data Acquisition Systems have a Software component and a Hardware component:

1) Hardware:

- Smart meters;
- Wide Area Measuring Systems;
- Power Transport and Distribution equipment;
- Communication network;
- Master Control Center;
- Master Terminal Unit;
- Remote Terminal Units and Programmable Logical Controllers;
- Data Concentration Units;
- Merging Units.

2) Software:

- Human Machine Interface;
- Database.

A. First generation or Monolithic SCADA

Independent systems with no connectivity to other systems, the main SCADA mainframe takes large amounts of data from the RTU's (Remote Terminal Units) via WAN (Wide Area Networks) in order to process it and acts as backup for the RTU's. With limited monitoring capabilities this system was impossible to upgrade due to the fact that every RTU manufacturer had its own proprietary protocol and introducing new equipment usually resulted in changing the entire system

B. Second generation SCADA or DCS SCADA

Distributed Control Systems (DCS) SCADA implies the sharing of control functions, distributed over a Local Area Network (LAN) that is used to connect multiple systems. This configuration has an increased processing power due to the distribution of different tasks on the Operating Stations. Compared to first generation SCADA where the backup system waits for the SCADA Mainframe to fail, DCS SCADA operates continuously and if one of the stations fail the workload is immediately picked up by another Operating Station without interrupting the entire process

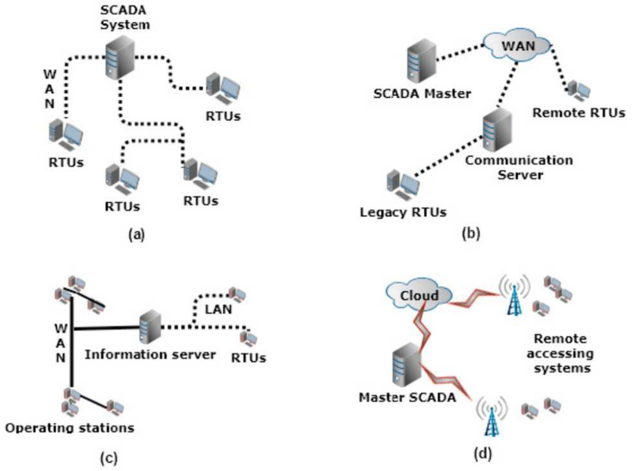


Fig. 3 SCADA evolution [1]

C. Third generation or Networked SCADA

Networked SCADA connects several parallel DCS SCADA under a single supervisor via WAN. This type of architecture incorporates Open-Source protocols which makes upgrading possible and facilitates the integration of large number of Edge devices. The most important advantage of Networked SCADA is the connection of distributed components using WAN specific protocols such as IP (Internet Protocol) instead of limiting all the SCADA system components to an isolated LAN network thus increasing security

D. Fourth generation or Cloud SCADA

Cloud SCADA system migration was made possible by the development of IoT (Internet of Things) and cloud computing systems. Compared to previous iterations of SCADA systems, cloud-based architectures offer the possibility to collect, process and store a wide variety of data in an efficient manner, with the help of interconnected servers. The processing power available for these systems facilitated the implementation of complex algorithms for control, thus enabling a full automation of the Power Energy Grid and helped integrate Smart Grid applications.

III. Cloud, Fog, Edge

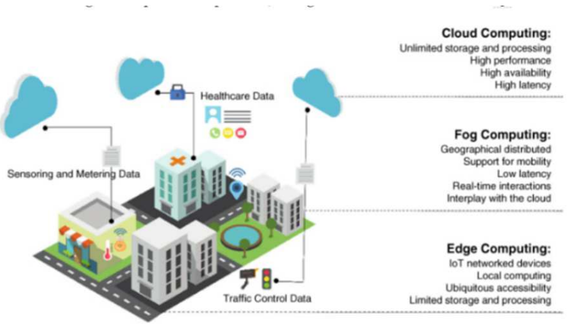


Fig. 4 Cloud, Fog and Edge Layers [2]

The term "Cloud" commonly refers to an infrastructure of computing services that are available on demand for the user, and offers a pool of resources (networking, storage, processing power and software services).
Cloud computing services are hierarchically structured, starting from the lowest level as follows:

- Servers installed inside the data center and the hardware resources associated (Processing power, memory, storage, bandwidth).
- Infrastructure as a Service (IaaS) - a pool of virtual machines (VM's) working together to create the virtual infrastructure.
- Platform as a Service (PaaS) – enables the creation and development of software solutions using IaaS resources.
- Software as a Service (SaaS) – provides ready-to-use software and applications for the business needs of cloud service customers.

Based on the implementation method, cloud systems are classified as follows:

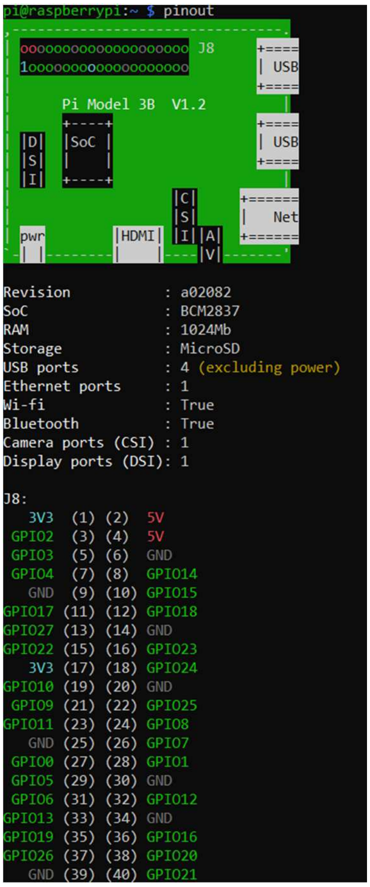
- Public Cloud – System that offers computation availability for the masses with no restrictions concerning accessibility.
- Private Cloud – In the case of a private cloud the data center is own by a private organization, and the data is available inside the organization only;
- Hybrid Cloud – Represents the extension of cloud computing with private, public computing techniques.

IoT integrations are commonly Cloud-centric (Cloud Centric Internet of Things) and introduce a gradual handling of data by bringing processing power closer to where sensor data is generated, commonly referred to as Edge computing. An intermediate processing layer in case the Edge layer is insufficient, is the Fog layer, offering additional resources, in order to send data to the Cloud.

IV. Implementation

The practical application includes a Raspberry Pi Model 3B V1.2 and Adafruit DHT 22 temperature sensor tied to Pin number 1 (3v power source), 6 (ground) and 7 (GPIO4). Data captured from the sensor was exported to a CSV, with the help of Python programming language.

Raspberry Pi is a low cost, small single board computer with similar capabilities as a desktop computer and uses standard keyboard and mouse. Raspberry Pi was developed in the UK by the Raspberry Pi Foundation for educational purposes.



```
pi@raspberrypi:~ $ pinout
,--------------------------------.
| oooooooooooooooooooo J8     +====
| 1ooooooooooooooooooo        | USB
|                             +====
|      Pi Model 3B  V1.2          |
|      +----+                 +====
| |D|  |SoC |                 | USB
| |S|  |    |                 +====
| |I|  +----+                     |
|                   |C|     +======
|                   |S|     |   Net
| pwr        |HDMI| |I||A|  +======
`-| |--------|    |----|V|-------'

Revision           : a02082
SoC                : BCM2837
RAM                : 1024Mb
Storage            : MicroSD
USB ports          : 4 (excluding power)
Ethernet ports     : 1
Wi-fi              : True
Bluetooth          : True
Camera ports (CSI) : 1
Display ports (DSI): 1

J8:
   3V3  (1) (2)  5V
 GPIO2  (3) (4)  5V
 GPIO3  (5) (6)  GND
 GPIO4  (7) (8)  GPIO14
   GND  (9) (10) GPIO15
GPIO17 (11) (12) GPIO18
GPIO27 (13) (14) GND
GPIO22 (15) (16) GPIO23
   3V3 (17) (18) GPIO24
GPIO10 (19) (20) GND
 GPIO9 (21) (22) GPIO25
GPIO11 (23) (24) GPIO8
   GND (25) (26) GPIO7
 GPIO0 (27) (28) GPIO1
 GPIO5 (29) (30) GND
 GPIO6 (31) (32) GPIO12
GPIO13 (33) (34) GND
GPIO19 (35) (36) GPIO16
GPIO26 (37) (38) GPIO20
   GND (39) (40) GPIO21
```

Fig. 5 "Pinout" command output from CLI

1) *The Kalaman Process*

- Predict next state [12]:

$$x(t) = F * x_{t-1} + B * u_{t-1} + \omega_{t-1} \qquad (1)$$

x(t) – state vector, which represents the output reading from the temperature sensor.
F – State transition matrix (F takes the value 1, assuming that the temperature does not vary quickly).
t – time.
B – Control input matrix (B takes the value 0).
u_{t-1} – Control vector.
ω_{t-1} – Process noise.

- Predict next covariance:

$$P(t) = F * P(t-1) * F^T + Q \qquad (2)$$

Process noise has covariance Q. Since the process variation is very low Q takes the value 0.05

- Compute the Kalman gain:
  $K = P(t) * \ H^T / (H * P(t) * H^T + R)$ (3)
  H – Measurement matrix.
  R- Observation noise has covariance R.
- Update the state estimate:
  $x(t) = x(t) + K * (measurement(t) - H * x(t))$ (4)
- Update covariance estimation:
  $P(t) = (1 - K * H) * P(t)$ (5)

2) *Output after running the algotritthm*

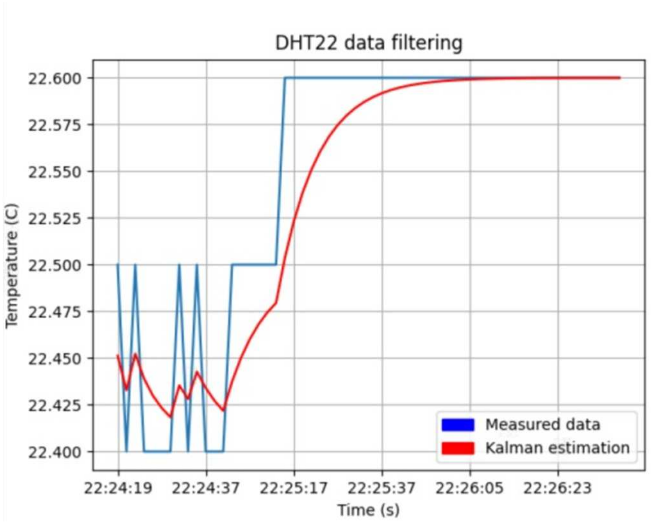


Fig. 6 Plot obtained after running Single State Kalman script in Python

V. Conclusions

With the addition of the Pi development board, processing power was brought closer to where the data is generated in order to normalize noisy measurements obtained from the sensor by applying the Kalman filter.

References

[1] Securing SCADA-based Critical Infrastructures: Challenges and Open Issues; DOI:10.1016/j.procs.2019.08.086;

[2] Fog and Edge Computing by Rajkumar Buyya; Satish Narayana Srirama Published by Wiley, 2019

[3] IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things by David Hanes; Rob Barton; Gonzalo SalgueiroPublished by Cisco Press, 2017;

[4] IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Thingsby David Hanes; Rob Barton; Gonzalo Salgueiro Published by Cisco Press, 2017

[5] Interoperability test for IEC 61850-9-2 standard-based merging units, Published in: 2017 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), https://ieeexplore.ieee.org/document/8086084

[6] https://www.cisco.com/c/dam/global/fi_fi/assets/docs/SMB_University_120307_Networking_Fundamentals.pdf

[7] Smart Grid: Communication-Enabled Intelligence for the Electric Power Grid by Stephen F. Bush, Publisher: Wiley-IEEE Press, Release Date: March 2014, ISBN: 9781119975809

[8] Smart Energy Grid Engineering by Hossam Gabbar, Publisher: Academic Press, Release Date: October 2016, ISBN: 9780128092323

[9] Power system analysis by K. N. Shubhanga, Publisher: Pearson, Release Date: May 2018, ISBN: 9789353063757

[10] Farzad Samie, Lars Bauer, Jörg Henkel, “IoT for Smart Grids: Design, Challenges and Paradigms, Chapter 2. Edge Computing for Smart Grid: An Overview on Architectures and Solutions” Springer, 2019, Pages 21-42

[11] Internet of Things for Architectsby Perry Lea Published by Packt Publishing, 2018

[12] Introduction to Kalman Filter and Its Applications By Youngjoo Kim and Hyochoong Bang, Published: November 5th 2018, DOI: 10.5772/intechopen.80600

[13] https://www.kalmanfilter.net/kalman1d.html