https://doi.org/10.1051/e3sconf/202125201051

A novel computing architect in ubiquitous electric IoT

Jie Bai[1], Hengtian Wu[2], Xing Wang[3], Dandan Guo[4] and Mengran Fang[1*]

[1] Research Institute, State Grid Information & Telecommunication Group Co., Ltd., Beijing, 100000, China
[2]State Grid energy research institute, Beijing, 100000, China
[3]Guangzhou HuanTV Technology Co., Ltd, Beijing, 100000, China
[4] Beijing Fibrlink Communications Co., Ltd., Communication Technology Center, Beijing, 100000, China.

Abstract. The Internet of things (IOT) has attracted great interest in many applications related to the monitoring and control of physical phenomena. The rapid popularity of Internet of things (IOT) equipment has brought unprecedented pressure to data communication and processing on communication network and central cloud infrastructure. Edge computing, as a potential platform to address such challenges, is facing significant challenges from the tension between resource constrained hardware and low latency applications. In this work, we propose a technical framework to meet the challenges. We suggest that computing and communication resources should be sliced horizontally to form a virtual computing platform to solve the problem of resource shortage.

1 The introducton of eIoT（electric Internet of Things）

To ensure the safe, stable and efficient operation of the state grid and improve the service quality of the state grid, sensors are increasingly applied in it, including measurement sensors and related sensors of power quality, power factor, phase, fault event, transformer and line load and other parameters. The Internet of Things is a combination of various information sensing devices and Information Communication Technology (ICT) resources to form physical entities with self-identification, perception and intelligent processing[1]. Internet of Things technologies include reliable information technologies for collecting, storing, analyzing, and processing massive amount of information as well as high-speed, two-way, real-time, and integrated communication technologies. The electric Internet of Things (i.e. eIoT) is the ubiquitous manifestation and application of the Internet of Things in the power industry. Generally, the eIoT system consists of four structures: sensing layer, network layer, platform layer and application layer[2]. It combines smart grid with artificial intelligence, modern communication technologies, cloud computing and data analysis technologies to realize ubiquitous connection and interaction. It is used for information collection, condition monitoring, feedback control, etc., to improve the depth and breadth of information in all aspects of the smart grid, and achieve business synergy, data penetration and open sharing.

2 Key enabling technologies in eIoT

In traditional centralized computing architecture, all data of the collected power terminals are transmitted to the service primary station, and the service primary station processes the feedback control information to the service terminal. With the application of cloud computing, big data, and new mobile communication technologies, the cyber-physical systemsof the power grid have evolved from simple automation to intelligent IoT system, providing important support for the production, scheduling, and operation of the power grid. The introduction of IoT technology has resulted in massive data transmission nodes, complicated routing, large transmission delay, low reliability, and high processing capacity for legacy primary stations. So New key enabling technologies have emerged to meet the challenge:

2.1Cloud computing model

In order to reduce the operation and maintenance pressure of IT equipment owners, virtualization technology as a key foundation of cloud computing enables the use of both physical and virtual resources for the different services in dynamic and flexible way as they are desired(it enables resource sharing among services and applications). In the cloud platform, all data are sent to the cloud for processing and most of the tasks are performed through centralized operation. As a result, the amount of data going into the deeper network will increase with uniform scaling such as 10 times and the latency increases in same way[3]. Therefore cloud computing will be a bottleneck of real-time processing for critical services. As device sensing is

[*]Corresponding author’s e-mail: fangmengran@sgitg.sgcc.com.cn

https://doi.org/10.1051/e3sconf/202125201051

local, it is desirable to keep sensed data processing and the corresponding decisions and actions to be local so that to reduce latency and improve privacy and security.

2.2Edge computing model

In recent years, mobile edge computing has developed rapidly. Mobile edge computing (MEC) makes computing closer to the data source network architecture to generate faster network service response, which can meet the higher requirements of the cyber-physical systems in smart grid such as real-time application, intelligent business, security and privacy protection and other aspects[5].

Edge computing utilizes the infrastructure closer to the user side and processes the data at the edge of the network, which not only improves the response speed of the system, but also reduces the system transmission requirements, and can be used as a complementary technology implementation of the current cloud system[4]. Edge computing enhances distributed intelligence through network edge processing. It is very suitable for such applications like automatic demand response services with a large number of terminals and high real-time requirements. It is therefore very suitable for applications in the power industry.

However, edge computing is facing with significant challenges posed by the tension between resource-constrained hardware and low-latency demanding applications due to the limitation of computing, storage and power supply resources of edge node.

3 A Novel Computing Arhitect

From the perspective of the edge computing of eIoT, we want computing and storage to sink as close as possible to the end device nodes, Despite continuous improvement of hardware, the endpoint device still remains a low-capable device due to the limitation of computing, storage and power supply resources （However, the end node device computing power and storage capacity are often limited, but also face physical constraints such as power consumption and cost.） . People need a novel computing architecture (we call it as horizontal slicing) to enhance the performance of terminal devices by leveraging the latest innovations in cloud computing and virtualization technology, where virtualization provides a critical foundation for network slicing by enabling the use of physical and virtual resources to create the services they design. As expected, with the increasing number of sensors and devices, as well as the significant improvement in the sensing capability of devices, a large amount of traffic has been generated on the smart grid. Since the device sensing is local, we hope to keep the sensing data processing and corresponding decisions and actions locally to reduce the waiting time and improve privacy and security. As a result, there will be less data going into advanced networks. It can be seen as using computing to help communication, that is, edge computing and processing reduce traffic to deeper infrastructure.

We expect horizontal slicing will be needed to help edge computation, e.g., local sever slices out a part of their computing resource to help computation at the robot devices, remote sever in cloud slices out a part of their computing resource to help the computation of the edge sever. Horizontal slicing helps devices to go beyond its physical limitation and is designed to enable edge computing offloading for accommodating the new trend of eIoT: The computing resources in the sever will be horizontally sliced, together with the computation resource slice of the devices (low-capable device) will be integrated to form a virtual computing platform through a communication link to significantly augment the computing capability of low capable devices as well as perform local traffic processing.

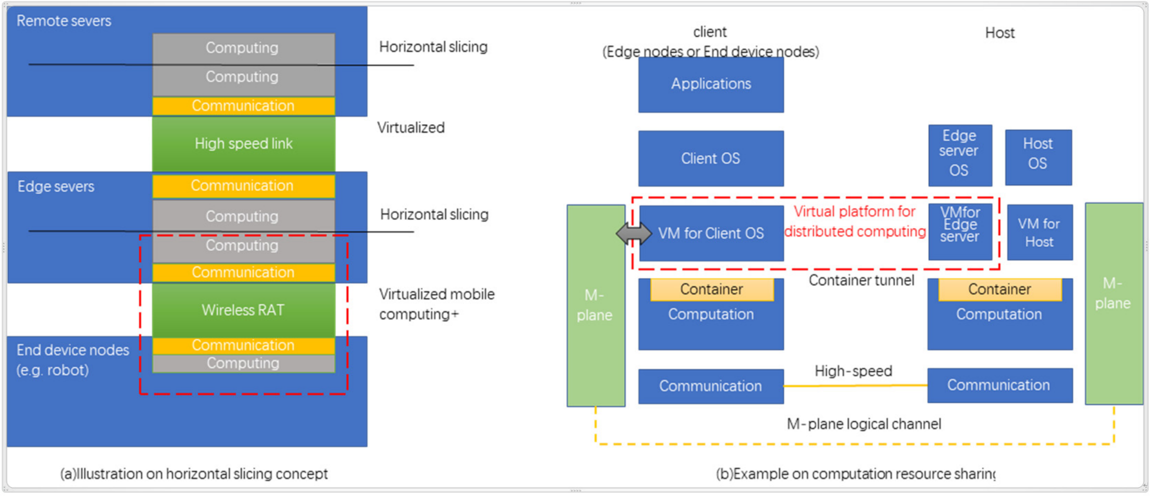

Figure 1. illustration of using horizontal slice to enhance the function of the device

The figure provides an example of the concept of horizontal slice. In order to optimize performance, we need to design the computing resource slicing and communication link supporting computing resource sharing (Figure (a) illustrates the concept of horizontal slice; figure (b) shows the example architecture of computing slicing and resource sharing. The main components are communication module, computing module, management plane (m-plane) module, virtual machine (VM) formed by virtualization technology and operating system (OS) running on VM. The calculation slice is managed by m-plane and implemented under OS. M plane monitors the resource usage and communication link condition of the system. When the M plane of the client sees the benefit of computing offloading (in terms of performance, cost, etc.), or when the client OS requests resources beyond the client computing capacity, it will request the host m plane to share the computing resources. If the host accepts the client's resource request, the client m plane informs the client of VM resource availability. Then, the client VM slices the slicable application according to the information from the M plane, and transmits the generated executable code to the container engine, and develops the container in the container engine to promote the distribution and execution of slicable applications. The container engine packages the sliced executable code into the container and transmits the packet that carries the container to the communication module. The communication module packages the container data and sends it to the host through the selected communication link. The communication module on the host decodes the received data block and then transmits it to the container engine. The host container engine unpacks the received container and hands over executable code to the edge server VM. When the calculation task is completed, the host's container engine packages the execution results into the container, and passes it back to the client container engine via a communication link. The client container performs the result unpacking. Then, the client OS application receives the execution result. Fig. 2 shows the described sharing process of computing resources. Note that in the example described, the compute task slice is implemented under OS. Alternatively, slicing can be implemented on OS or later. In this case, container information can be considered as normal user traffic and no special design is required in communication links. However, performance (such as processing latency) will be affected by operating system performance. Slicing on operating systems lower than OS is expected to achieve better performance. In this way, the host computing resources are virtualized into two parts. Part of it is used by your own application and runs on your own OS. The other part is assigned to the client and is used as a remote resource by the client. The application on the client must be slicable to execute simultaneously on both VMS. One is on the client and the other is on the host. The client VM can act as the primary VM, and the host VM can serve the client master VM as a co processing engine. The business hosting the container can be made visible to the communication link.

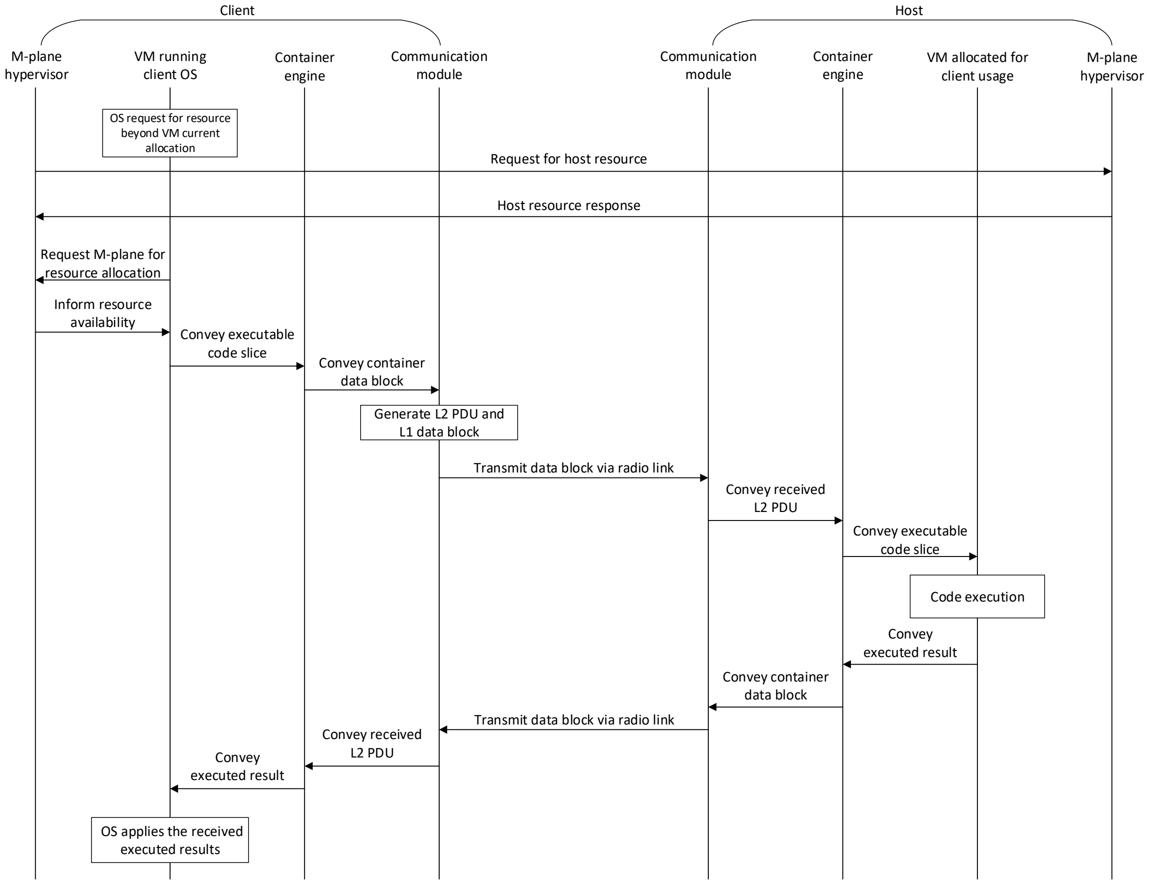
Client
Host
M-plane hypervisor
VM running client OS
Container engine
Communication module
Communication module
Container engine
VM allocated for client usage
M-plane hypervisor
OS request for resource beyond VM current allocation
Request for host resource
Host resource response
Request M-plane for resource allocation
Inform resource availability
Convey executable code slice
Convey container data block
Generate L2 PDU and L1 data block
Transmit data block via radio link
Convey received L2 PDU
Convey executable code slice
Code execution
Convey executed result
Convey container data block
Transmit data block via radio link
Convey received L2 PDU
Convey executed result
OS applies the received executed results

Figure 2. Computation resource sharing procedure

4 Application Examples

The following is a potential application of a novel computation architect for patrol robot in smart grid

Most of the current data analysis of patrol robot is executed in offline way, which has poor real-time performance. Only a few devices have the real-time detection ability of static image, but it is unable to directly analyze and identify the video stream locally due to high complexity, consequently the localized real-time defect detection for video stream acquisition by UAV has not yet been realized. The current robot inspection architecture is shown in the figure as below. Patrol robot collects and stores video locally and then copy it to server for offline analysis via USB or Ethernet.

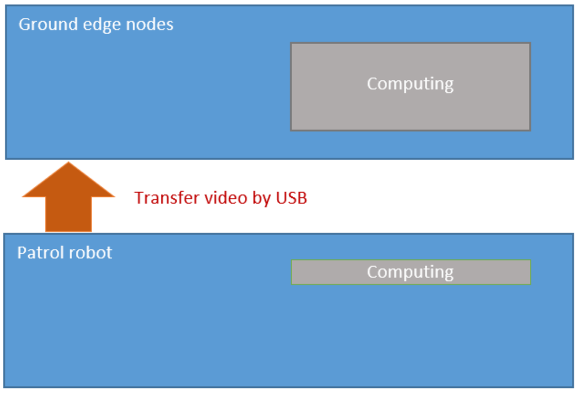

Figure 3.The current robot inspection architecture

The application of a novel computation architect in UAV inspection can construct two horizontal computing platforms by using high-speed communication links e.g. high-speed 5G link between local edge nodes and UAV/remote sever to realize real-time defect detection of UAV video streams. The architecture is shown below.

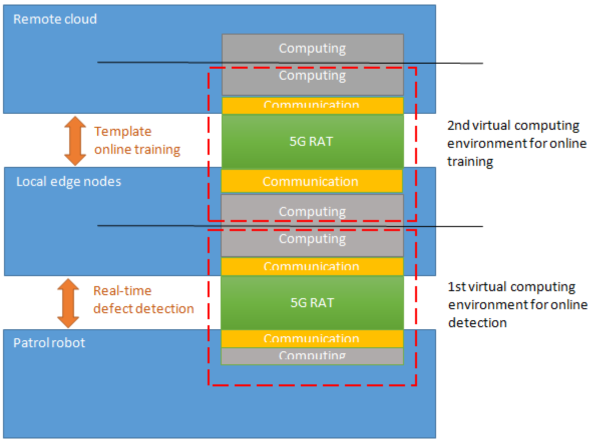

Figure 4. A novel computation architect in Patrol Robot

Object detection in the patrol robot is done on the first virtual computing platform formed by part of local server's computation resource whilst the online training is performed on the second platform constructed by local edge sever and part of remote cloud server via high-speed 5G link. These two computing platforms go across and beyond physical boundaries of different nodes targeting to improve the performance of the low capable devices.

5 Conclusion

With the development of power wireless private network, 5G continues to evolve, applying new mobile edge computing technology to strong smart grid and ubiquitous IoT construction, and building a strong information platform with cloud platform and physical network architecture to promote grid enterprises. The improvement of the comprehensive energy service level provides the society with a safe, reliable, efficient and green energy supply; further ensuring energy security and promoting the sustainable development of society.

References

1. Wenda Tang,etc,"An Offloading Approach in Fog Computing Environment", IEEE SmartWorld2018.
2. Dan Wu, etc., “Dynamic Distributed Resource Sharing for Mobile D2D Communications”, to appear in IEEE Transactions on Wireless Communications2015.
3. Feng Ye, etc, “A Big Data Driven and Cloud Computing Based ICT Framework for Smart Grid”, submitted to IEEE Communications Magazine2015.
4. Qun Wang,etc.:Hierarchical Collaborative Cloud and Fog Computing in IoT Networks. WCSP 2018: 1-72017
5. Zhenya, Liu. A&Q for Smart grid knowledge. Perking: Chinese Power Press, 2010