Cloud-Based Information Infrastructure for Next-Generation Power Grid: Conception, Architecture, and Applications

Fengji Luo, Member, IEEE, Junhua Zhao, Member, IEEE, Zhao Yang Dong, Senior Member, IEEE, Yingying Chen, Member, IEEE, Yan Xu, Member, IEEE, Xin Zhang, and Kit Po Wong, Fellow, IEEE

Abstract—This paper gives a comprehensive discussion on applying the cloud computing technology as the new information infrastructure for the next-generation power system. First, this paper analyzes the main requirements of the future power grid on the information infrastructure and the limitations of the current information infrastructure. Based on this, a layered cloud-based information infrastructure model for next-generation power grid is proposed. Thus, this paper discussed how different categories of the power applications can benefit from the cloud-based information infrastructure. For the demonstration purpose, this paper develops three specific cloud-enabled power applications. The first two applications demonstrate how to develop practical compute-intensive and data-intensive power applications by utilizing different layered services provided by the state-of-the-art public cloud computing platforms. In the third application, we propose a cloud-based collaborative direct load control framework in a smart grid and show the merits of the cloud-based information infrastructure on it. Some cybersecurity considerations and the challenges and limitations of the cloud-based information infrastructure are also discussed.

Index Terms—Cloud computing, distributed computing, smart grid.

I. INTRODUCTION

Driven by the new emerging technologies and the increasing pressure of the global warming, the conception of “smart grid” was proposed in the last few years [1]–[3]. According to the definitions given by the National Energy Technology Laboratory of the USA [1], smart grid should have the features of self-healing, attack tolerance, running more efficiently, motivating demand-side management, high penetration of the renewable energy, etc. These features make the future grid to be a complex cyber-physical system and impose some significant requirements on the information infrastructure of the next-generation power grid, including fast reaction to disturbances and faults, wide area data management, high-performance computing and real-time analysis, and data security.

A. Limitations of Current Information Infrastructure

Currently, the information structure of the grid adopts the centralized structure, where the control entities of a utility are directly connected to the energy management system (EMS) and the EMS acts as the main control center. The control activities are also performed in a centralized manner. The remote terminal unit of each substation interacts with the EMS to report the state information of the underlying power devices.

This centralized information infrastructure cannot satisfy the information requirements of the future grid. The first limitation is that the control center can hardly manage the massive data generated by future grid due to its storage bottleneck. For instance, in the future grid a phase measurement unit (PMU) generates 50 or 60 phasor measurements per second [4]. Such big data cannot be fully transmitted to the control center due to the bandwidth limit. It can be expected that even a small number of PMU will generate a large amount of data and reach the bandwidth bottleneck. Even the bandwidth can be upgraded to serve the peak-rate data transmission (obviously this is not a cost-efficient solution), the storage capacity limitation of the control center restricts the storing of the big data centrally. Therefore, to effectively aggregate the wide area distributed data, a decentralized, platform-centric infrastructure is desired.

With the exploration of the big data, the computing power consumption of the future grid will become a big challenge for the centralized information infrastructure. In order to be self-healing and fast react to online disturbances, the control center needs to process the big data quickly. Also, many online/offline applications will run on the servers simultaneously, which imposes heavy load for the computational resources of the control center. A new platform-centric information infrastructure is thus needed to deliver scalable and reliable computational powers to the future grid.
Today’s centralized structure is also vulnerable to the cyber-attacks. For example, one attacker may attack the transmission channel between the EMS and the control entity by injecting a large number of malicious data packages to make the congestion. Since the current information infrastructure is a star network, the significant data transmission delay is unavoidable under this attack. Or the attackers may impose distributed denial of service (DDoS) attacks to breakdown the control center server. To ensure the reliable and secure operation of future’s grid, a robust information infrastructure is needed.

### B. Related Works and the Scope of This Paper

There have been a few initial works discussing the vision of applying cloud computing on future power grid. Luo et al. [5] presented a prototype of hybrid cloud computing platform for smart grid and discussed some potential cloud-based smart grid applications; Luo et al. [6] gave a conceptual design of the power cloud data center to deliver multilayer services to smart grid; Bitzer and Gebretsadik [7] discussed how smart grid applications can benefit from cloud computing framework, and described a cloud computing platform for supervisory control and data acquisition (SCADA) data monitoring application; Mohsenian-Rad and Leon-Garcia [8] proposed a coordination dispatch algorithm for cloud computing and smart power grids. The power flow constraints are incorporated into the cloud datacenter dispatch algorithm, so that the impact of the dispatch of computational resources on power grid is considered; Rusitschka et al. [9] analyzed the limitation of the data management infrastructure of the current grid, and then proposed a cloud framework for smart grid data management; Kim et al. [10] established a cloud-based demand response application for smart grid, where the utility negotiated load shedding (LS) price with the large number of energy consumers.

Most of above works only described general models of the cloud platform for power grid, and summarize some conceptual applications. The major contributions of this paper are twofold.

1. **This paper proposes an explicitly layered model of the cloud computing-based information infrastructure for the next generation power grid.** Just as the well-known-layered model of the Transmission Control Protocol/Internet Protocol (TCP/IP) protocol, the layered cloud-based information infrastructure model explicitly decouples different cyber-physical entities and identifies their roles.

2. **This paper discusses how power applications can benefit from the cloud-based information infrastructure and develops three specific cloud-enabled power applications.** The first two aim to demonstrate how to develop compute-intensive and data-intensive power applications on the state-of-the-art public cloud platform while the third one develops a cloud-based collaborative direct load control (DLC) architecture based on the proposed layered model.

The reminder of this paper is organized as follows. In Section II, the proposed cloud computing infrastructure is introduced. In Section III, the cloud-enabled power applications are studied. Section IV discussed some cybersecurity issues of the cloud-based information infrastructure. Section V discussed the potential challenges and limitations for the cloud-based information infrastructure. Finally, the conclusion is drawn in Section VI.

### II. LAYERED MODEL OF THE CLOUD-BASED INFORMATION INFRASTRUCTURE FOR THE FUTURE GRID

#### A. Basic Conception of Cloud Computing

According to the definition of Foster et al. [11], cloud computing is a large-scale distributed computing mode which can form a virtualized, dynamically scalable resource pool to deliver services to users on demand through the Internet. As a new computing mode, cloud computing is an integration of multiple major technologies (Fig. 1), including grid computing, hardware virtualization, utility computing, Web service, and automation computing. The more details of cloud computing can be referred to [12].

Supported by these core technologies, cloud computing can deliver different level of services to users.

1) **Infrastructure as a Service (IaaS):** IaaS offers virtualized resources to users in an on-demand manner. Users can access and use the virtualized resources directly on demand.

2) **Platform as a Service (PaaS):** PaaS offers a programmable environment where developers can construct applications directly on the cloud side.

3) **Software as a Service (SaaS):** The complete developed software is delivered to users through cloud portals. SaaS makes users shift from locally installed computer programs to online software services that offer the same functionality.

#### B. Layered Information Infrastructure Model

In the future grid, the information flow and decision-making process can be supported by the cloud-based information infrastructure, in all of the generation, transmission, and distribution sides [6], [9]. The proposed layered model for the information infrastructure of the future grid is shown in Fig. 2.
A well-designed layered model often decouples the dependencies of different elements and defines the role boundaries of each layer. A well-known example is the layered Open System Interconnection (OSI) model of the TCP/IP protocol [13]. In our model, each layer relies on underlying layers and only communicates with its neighboring layers.

1) Power Fabric Layer: The power fabric layer includes the wide-area physical resources of the power grid. Those resources are involved in the generation side (power plants, wind turbines, etc.), transmission side (transmission lines, transformer, etc.), distribution side [substations, feeders, electric vehicles (EVs), etc.], and other control elements. This layer forms the foundation of the power grid.

2) Sensor and Actuator Network Layer: The sensor network layer monitors the state of the power resources and collects the raw data. The elements of this layer may include SCADA system, PMU, and smart meters. In addition to the hardware sensors, the software and services which generate data for some specific objectives are also classified into the sensor network (e.g., the wind speed forecasting system).

3) Cyber Fabric Layer: This layer consists of the cyber resources, including computational resources (servers, workstations, PC clusters, etc.), storage resources (disks, database, data warehouse, etc.), software systems, and communication network infrastructure. Here the cyber resources may be invested and owned by different parties, such as the public cloud service providers.

4) Secure Communication Layer: This layer encrypts and authenticates the data to ensure it cannot be read or interfered during communication. Different secure communication technologies can be applied in this layer. For example, the data encryption can be implemented by establishing the secure channels, where each channel is secured by the session key. The communication parties assign the key to the secure channel through the key exchange procedure [14]. For establishing the secure channels, the public key infrastructure can be used.

5) Dynamical Scheduling Layer: This layer implements the dynamical scheduling algorithms for the cyber system performance optimization. Computational tasks are scheduled among different computational resources to achieve high-performance computing efficiency and optimize the computational load balance of the computational resources. Critical and large volume data of the power grid are storage and cached distributed.

6) Firmware Layer: Firmware layer performs the hardware virtualization for the cyber resources. It dynamically generates the virtualized machines with different scale and configurations. It also maintains the life cycles of the virtualized machines and does low-level scheduling tasks, such as mitigating the operation environment from one server to another when physical fault happens. The firmware layer also provides authenticate and access control services to ensure the secure access of the virtualized machines.

7) Abstracted Service Layer: This layer provides various power grid-oriented services in different cloud service layers (IaaS, PaaS, and SaaS). In the IaaS, it delivers the virtualized machines on demand to power grid participants; the PaaS provides open application programming interface (API) to power grid participants to develop different applications; the SaaS provides online services such as data analytics, stability analysis, and system modeling, to support decision-making of power grid participants.

8) Power Application Layer: Based on the Web service interfaces of the cloud-based information infrastructure, power grid participants can develop various applications, ranging from local to wide-area operation and control.

C. Merits of the Cloud-Based Information Infrastructure

Based on the aforementioned architecture, we summarize some merits of the cloud-based information infrastructure for the power grid, which are significantly different with the current centralized information infrastructure.

1) Scalable and Elastic Virtualized Resource Provision: By utilizing the hardware virtualization technology, the firmware layer can create virtualized resource instances on top of the hardware resources. Each virtual machine can run its own operation system and software stacks. Backed by the massively powerful physical resources in the cyber fabric layer, the firmware layer can provide nearly unlimited capacities of virtualized resources to the power users, and it can dynamically scale the resource provision according to the different power applications requirement.

2) Distributed Power Data Management: In the cloud-based information infrastructure, the dynamical scheduling layer will aggregate and manage the wide area power data. The advanced data dissemination, routing, and replication methods
implemented in this layer can support the fast data transferring and retrieving by the power control centers and other different smart grid participants, and optimize the performance of the whole underlying data storage network. The higher-level data management services and data analytical tools implemented in the Web service layer can let the power users effectively manage and analyze the underlying large-scale data to optimize the operation of the grid.

3) Scalable High-Performance Computing: The elastic computational resource provision feature of the cloud-based information infrastructure enables the high-performance processing for different scale of the power applications. Such powerful computing capacities can hardly be achieved by the centralized computing mode. Also, the fast parallel processing capability of the cloud infrastructure can effectively support the power users to do the real-time analysis and make online decisions.

4) Security and Fault Tolerance: Compared with the centralized structure, the cloud-based information infrastructure is more robust and fault tolerant. The secure communication layer can enhance the data security and integrity of the communications. The authentication and access control services in the firmware layer and the secure computing services in the Web service layer can ensure the confidentiality of the users. And many cyberattack defending techniques for the specific power applications can be implemented in the abstracted service layer and the power application layer to enhance the cybersecurity of the grid.

The hardware virtualization technology in the firmware layer can significantly improve the fault-tolerance of the system. Given there is any hardware failure occurs, the virtualized resource management schemes in the firmware layer can mitigate the virtualized resources images to other hardware to avoid the single point breakdown. Also, the data and task scheduling mechanisms in the dynamical scheduling layer will enhance the robustness of the whole system.

5) Cost Reduction: In the cloud-based information infrastructure, the cyber hardware is maintained in the cloud side, and the services are delivered to the users in the on-demand manner. By mitigating the applications to the cloud side, the power users only need to pay-on-demand and thus can largely reduce the local investment and operation costs on the hardware and software. This will also encourage the information sharing and application integrity of different power entities, and promote the economical operation of the grid.

6) Access the Cloud at Any Time and Anywhere: Since the communication media of the cloud computing is the Internet, the power users can access the cloud through the Web service interface as they want, by using various lightweight media (tablet, smart phone, etc.)

D. Implementation Technologies

After introducing the architecture of the cloud-based information infrastructure, in this section we will give some overview discussions about the implementation technologies of the different layers of the infrastructure. The technologies discussed in this section can be used as “bricks” to construct the cloud-based information infrastructure. It also should be mentioned that the technologies introduced below are just some representative technologies summarized by us. More technologies can be found in both of the industry and academic circle. And since each layer in our layered model represents an open research area, more useful tools can be developed in future.

In the secure communication layer, the Rivest-Shamir-Adleman (RSA) public-key-based encryption method [15] has been widely applied in most of Internet-based communications. The theoretical foundation of the RSA is the decomposition of large prime number, which can therefore fundamentally resist the eavesdropping of the cyberattackers. In addition to RSA, there are also some other data encryption technologies which is promising to be widely deployed in the foreseeable future. One significant technology is the quantum cryptography technology [16], whose theoretical foundation is the quantum mechanics. Instead of transmitting bits, in quantum cryptography the data are encoded as qubits. Qubits are polarized by different directions. The quantum key distribution starts by sending a large number of qubits from the sender to receiver. If a eavesdropper wants to get the information of any qubit, he or she will have to measure it. Since the eavesdropper knows nothing about the qubit’s polarization, he or she can only measure it by guess, and then send another randomly polarized qubit to the receiver. But all these operations will be discovered at the end of quantum key distribution. After a brief communication, the key will be identified to be safe or not.

In the dynamical scheduling layer, there are several technologies can be applied in the computational load scheduling. For example, cloud scheduler [17] is a commercial software system for scheduling the tasks among the cloud resources. Many researchers also proposed some computational load scheduling strategies. Selvarani and Sadhasivam [18] proposed a task scheduling scheme to schedule the task groups in cloud computing platform, where resources have different resource costs and computation performance. Tsai et al. [19] proposed the cost and time models for the computational tasks in the cloud computing environment, and studied an optimal task scheduling model. Some other works can be referred to [20]–[23]. There are also some technologies for the distributed data management in the cloud environment. Brocade Vyatta virtual router [24] delivers advanced routing for the data packages in physical, virtual, and cloud networking environments. It includes dynamic routing and policy-based routing in a platform that is optimized for virtualized environments. The HP VSR 1000 router series [25] is a virtualized application that provides functionality similar to a physical router. It enables significant operational savings as a result of its agility and ease of deployment. Spanner [26] is a data synchronous replication system adopted by Google’s cloud infrastructure, which stores data in multiple data centers to avoid the data lost in the case of the hardware disaster. Wei et al. [27] proposed a cost-effective dynamic data replication scheme. By adjusting replica number and location according to workload changing and node capacity, this scheme can dynamically redistribute...
workloads among data nodes in the heterogeneous cloud. Sun et al. [28] proposed a dynamic data replication strategy to improve the data availability in the cloud computing environment. Another promising technology is the data-centric data dissemination and routing technology [29], [30]. This technology is based on a simple observation: the content of the data is more important than the identity of the node that gathers them. In the data-centric data dissemination and routing technology, the identity of the data storage node is less relevant, and the data are named and routed referring to these names rather than the storage node addresses. Users can define certain conditions to query the data. Certain name-based data routing algorithms are often adopted.

In the firmware layer, some mature industrial products can be applied to perform the virtual resource management. For example, the Dell virtualization management solutions [31] can provide a series of services to users, including performance and availability monitoring, and capacity planning for the resources. Right scale [32] is a platform for managing and deploying cloud resources across public and private environments, providing users the tools to configure, monitor, automate deployments, and govern controls and access. VMware Workstation [33] provides a suite of software solutions for creating and managing the virtualized resources and constructs the cloud data centers.

In the abstracted service layer, many sophisticated services can be developed. There have been many industrial products which can be deployed in this level. These products provide higher-level services in many aspects. As a SaaS-level product, Google Cloud SQL [34] runs on the Google cloud data center. It allows users to perform higher-level data management in the cloud. Amazon elastic compute cloud (Amazon EC2) [35] provides the IaaS-level services by delivering the virtualized server instances directly to the users. CloudFuzion [36] provides the PaaS-level services on which the users can develop and fast solve different engineer problems. It also provides a suite of power grid application solutions. MapReduce [37] is a parallel data processing framework which has been widely used in the cloud platforms. It can parallel process parallelizable problems across huge data sets using a large number of computing nodes. A famous software implementation of the MapReduce framework is the Apache Hadoop [38], CloudBroker [39] is a software which provide higher-level task scheduling services among the resources belonged to different cloud providers. Some other products (such as Cloudyn [40] and Cloud Cruiser [41]) can provide some ancillary services such as cost tracking, resource monitoring, and task tracking.

Supported by the powerful cloud virtualized resources, some other promising technologies can also applied in the abstracted service layer. For example, stream computing is a software framework designed to support the fast real-time stream data processing. It can be seamlessly deployed on the cloud resources to do the fast data flow process. The widely used stream computing software include Storm [42], S4 [43], and StreamBase [44]. The secure multiparty computation (SMC) [45] is another technology which is promising to be widely applied to enhance the data security in the cloud environment. SMC addresses the data security issue in the collaboration. SMC makes it possible to perform the collaborative computations directly on the encrypted data without decrypting it. The theoretic foundation of SMC a data encryption technique called the homomorphic encryption [46]. The homorphic encryption technique allows the encrypted data to be processed by certain operators (additions and multiplications), and the results are the same with that of processing the plain data. Based on this, the basic idea of the SMC is that given a computation function, a set of specific protocols can be designed between the collaborators to enable them to only share data encrypted with homomorphic encryption and use the encrypted data to do the computation so that each party can only decode the computation result, rather than others’ contributed data.

III. APPLICATIONS

This section discusses how different power applications can benefit from the cloud-based information infrastructure. First, we identify four categories of the power applications, and discuss how they can be effectively solved by cloud computing. Followed by this, we give three specific technical applications for the demonstration purpose. The first two demonstrate how to build compute-intensive and data-intensive power applications by utilizing the practical cloud platforms, respectively; in the third one, we develop a cloud-based architecture for a collaborative smart grid application to show how smart grid applications can benefit from the proposed layered model.

A. Enabling Power Applications by the Cloud-Based Information Infrastructure

Relying on the powerful computing capability and layered functional services, the cloud-based information infrastructure can be adapted to various kinds of power applications. In this section, we summarize the power application as four categories, and each of which covers some common computational features.

1) Compute-Intensive Power Applications: The compute-intensive power applications represent the power applications which needs a lot of computations. This category of power applications is very common in modern power system analysis. For example, a complex large-scale transmission network planning problem often requires a long-time optimization process (several hours to even several weeks) [47]. Another example could be the system stability analysis, where the “N-1” or even “N-2” fault analysis [48] is often performed to study the post-fault stability of the grid. For a large system, this will lead to a huge number of fault scenarios. Other compute-intensive power applications include voltage stability analysis [49], unit commitment [50], and distribution network optimization [51].

In order to get the compromise between the accuracy of the analysis result and the computation time, the power engineers often need to do some numerical approximations and relax the complexity of the analysis method (e.g., reduce the iteration number of the optimization method and reduce the number of the analysis scenario). In many cases, such compromise will affect the achievement of the optimal solution.
In order to accelerate the compute-intensive applications without significantly sacrificing the analysis result accuracy, the high performance computing (HPC) resources are also applied, such as the multicore server or multinode cluster [52], [53]. There are some limitations for these traditional HPC approaches.

1) The software configuration and hardware maintenance of those HPC resources are often time-consuming and need the expert knowledge.

2) The investment and maintenance costs of those HPC resources are often very large, which makes them hardly accessible or affordable for small-scale research and industrial organizations, specialized applications purposes or short-term projects.

3) There lacks of the standard open API for those local HPC resources, which makes them hardly to integrate with existing systems and processes.

4) The local HPC resources are often operated at capacity limit. There is no elastic service providing mechanisms for them to operate in a cost-effective and energy-effective manner.

The power industry can benefit a lot by running the compute-intensive applications on the “cloud.” First, since the communication media of cloud computing is the Internet, the power users can immediately access to the information infrastructure in on-demand manner; second, the power users can easily operate the HPC resources in the cloud, which are not existed in-house; third, supported by the large-scale computational resources, the cloud infrastructure can provide nearly unlimited capacity to applications, which allows the power users to quickly solve their problems without sacrificing the computation accuracy. And the resource provision of the HPC resources is virtualized and elastic (implemented in the firmware layer in the proposed layered model), meaning that the power users can dynamically scale the utilization of the resources based on the problem scale, budget, and other practical considerations. Fourth, all the resources are managed by the cloud side, which reduces the investment and operation costs of the power users to the minimal level.

2) Data-Intensive Power Applications: Data-intensive power applications represent the power applications which have strong demand in storing, managing, and analyzing data. Based on the workload of the data analysis, a data-intensive power application would also be compute-intensive. With the data explosion of the modern power systems, data-intensive applications have increasingly important roles [54]. For example, a machine learning-based system security assessment and control needs to learn knowledge from a large number of high-dimensional system operation points to predict the security of the system [55]. Another example is that in the smart grid, an electricity retailer would aggregate the big data of the energy consumption records of the end users, and analyse the big data to find the energy consumption patterns of the users [56]. Based on it, the retailer can optimize the retailing price schemes for each individual user, so as to optimize the operation of the distribution system. Other data-intensive power applications include data-driven wind power forecasting [57], data-driven wind turbine allocation [58], data-driven battery energy storage system capacity determination [59], nonintrusive appliance load monitoring [60], and fault diagnosis [61].

Traditionally, the development of the data-intensive applications are often developed from scratch by the power engineers and researchers, and the data analysis is often performed in the local resources. This pattern is costly, low-efficient, and can hardly cater for the big data requirements in the future grid. With the widely deployment of the various high-frequency data sources (e.g., the distributed renewable sources, PMUs, and two-way communication infrastructure), the wide-area data aggregation and analysis can hardly be performed in the centralized pattern. With the cloud-based information infrastructure, the data-intensive power applications can be significantly strengthened in three aspects.

1) Unlimited Data Aggregation and Management Resources: Just as the HPC resources, the firmware layer of the layered model can elastically provide data collection and storage resources on-demand, which can well satisfy the big data management of the future grid. In addition to the physical resources, the advanced data management services in the abstracted service layer (one example is the Google Data Store [49]) can be utilized to support the transparent retrieving and querying of the wide-area data.

2) Advanced Data Network Performance Control: The distributed data management techniques in the dynamical scheduling layer will effectively optimize the performance of the data network. For example, the data package routing and replica techniques can dynamically optimize the transfer routes of the data packages and the locations of the replicas, so as to fast response to the users’ requests as well as maintain the robustness of the data network.

3) Advanced Data Processing Framework: As an important part of cloud computing, the big data processing frameworks can make the power engineers and researchers effectively develop the data-intensive applications. These frameworks can be seamlessly integrated with the virtualized resources in the cloud, and can be easily scaled to parallel processing the large datasets. A famous big data processing tool is the MapReduce framework and its implementation software Apache Hadoop.

3) Collaborative Power Applications: The third category of power applications we discuss here is the collaborative power applications. With the development of smart grid and the ongoing trend of deregulation, there will be many applications where multiple parties need to collaboratively accomplish a specific task. A simple example would be the bidding process of the electricity market, where different generation companies submit the bids to the independent system operator (ISO), and the ISO matches the bids to determine the market clearance prices [62]. Another example is the EV dispatch in a distribution system. The EV owners submit their individual driving demands to the system operator (e.g., the estimated driving distance, plug-in time, plug-out time, and expected state of the charge). By gathering such information, the system operator decides the optimal charging plans for the EVs, and
sends the plans to the EV owners [63]. Other collaborative power application includes the cooperative dispatch of virtual power plant resources [64], and agent intelligence-based demand response [65].

A collaborative power application could be compute-intensive and/or data-intensive. Or it could be neither compute-intensive nor data-intensive. Anyway, some basic requirements can be drawn from the collaboration process.

1) Scalability: The collaboration should be scalable to support different number of collaboration parties.

2) Security and Confidentiality: When multiple parties share data in the collaboration process, the secure data access and authority are desired. And the confidentiality of the data should be ensured during the whole collaboration process, meaning that the private data should not be revealed to the irrelevant parties.

3) Efficiency: The collaboration should be efficient and fast.

By applying the cloud-based information infrastructure to manage the collaboration process, the cloud platform will act as the collaboration coordinator, which is responsible for hosting the collaboration system (data and computation), as well as managing and enforcing the agreed access policies. The firmware layer can provide the access authentication services and perform the policy-based access control. In addition to this, the services provided by the secure communication layer and the secure collaboration techniques (such as the SMC technology) in the abstracted service layer can significantly keep the security and data confidentiality of the collaboration process.

By running the collaborative power applications in the cloud side, the elastic virtualized resource provision capability of the cloud-based information infrastructure can also ensure the high scalability and efficiency of the collaboration process.

4) Real-Time Power Applications: The future power grid is expected to have the ability of self-healing and fast respond to various disturbances, which drives the establishment of the different kinds of real-time operation and control applications. Here although a real-time power application can also be one of above three categories, an application which is classified as one of above three categories is not always be a real-time power application. For example, a long-term network planning application could be compute-intensive, but in most of cases it is considered as an offline task. Another example is that the large-scale EV dispatch problem is a typical collaborative application which requires a large number of information exchanges among the system operator, load aggregator, and EV owners, but the dispatch is often performed in the day-ahead stage, based on the various forecasted information [63].

A noticeable feature of the real-time power applications is the capturing and fast processing of the real-time data streams. The processing of the online data might also be organized as a complex work flow, where multiple parties can be involved. From this, we can summarize two main requirements of the real-time applications.

1) High Processing Speed: For the real-time applications, the power operators (or the software agents) often need to make decision and take control actions in a short time, thus the high computing speed is required for processing the real-time data streams, which are often with noises and unpredictable arrival rates.

2) Fault Tolerant and Reliable Service Provision of the Computing Environment: The efficient operations of the real-time power applications need to be supported by the robust computing infrastructure to prevent the underlying hardware breakdowns from halting the online data processing work flow.

A typical example of the real-time power application could be the real-time system peak load tracking [66]. With the fine-gained data collection of the SCADA system and the widely deployment of the AMI, the volume of the real-time data generated is huge. The current data collection rate of the SCADA system and AMI are every 3–5 s and every 15 min, leading to the gigabyte-level cached data for a regional distribution network. In this context, the system operator needs to analyze the data to track the system peak load and take appropriate control actions. Other real-time power applications include the system restoration [67], real-time frequency regulation [68], and predictive control-based energy management of the buildings [69].

The real-time processing of such vast amounts of data can be a big challenge of the traditional computing modes. As an integrated technique of cloud computing, stream computing (located in the abstracted service layer) provides a highly flexible and effective solution for the online data processing and decision-making. In the stream computing architecture, by properly designing the data flow processing topology, the real-time data generated by the data sources can be collected by the distributed nodes. The data flow then can be sent to the other processing nodes to do step-by-step further analysis, and processed data flow can be finally sent to the target node to support the decision-making. And the advanced data processing framework (such as the MapReduce) could be seamlessly integrated with the stream computing platform to accelerate the data processing.

The real-time power applications will also significantly benefit from the virtualization technology of the cloud computing, which is located in the firmware layer of the proposed layered model. The virtualization technology provides an ideal solution for addressing the hardware exception problem. When the underlying hardware exception occurs, the virtualized resource management system can mitigate the virtualized computing instances on which the real-time applications run to other physical resources, so as to ensure the secure and effective running of the real-time power applications.

B. Compute-Intensive Application Demonstration:
Cloud-Enabled LS Scheme for Voltage Stability

In [49], we developed a parallel processing framework for the event-driven LS against voltage collapse, and deploy it in a local PC cluster. The parallel process can be easily mitigated into the cloud side by utilizing the IaaS-level services, where users can apply for the virtualized resources and operate them just as local resources.

1) Problem Overview: When the voltages of the buses of a power system violate the allowable scopes, the data-driven
LS strategy is performed to shed the power loads on different buses, so as to restore the voltages. The LS problem can be modeled as an optimization problem, where the decision variable is the shed load vector, representing the shed load amount of each bus. The optimization objective is to minimize the total interruption cost of LS. The model is also subjected to the voltage stability margin constraint, voltage limits constraint, branch current limits constraint, and the power flow equations constraint. The details of the problem can be found in [49].

To support online application, the LS decision-making process must be performed very fast to prevent the system from voltage collapse. Therefore, the parallel-differential evolutionary (DE) algorithm is employed to solve the model. The parallel strategy follows the master–slave pattern. The master process divides the whole population into \( N \) subpopulations and distributes them to the slave processes. The slaves connect with each other through the master in a predefined topology such as a “ring.” Then each subpopulation evolves separately in a parallel pattern. To promote the information sharing among the subpopulations, in each iteration, the best individual of each subpopulation is moved to the next subpopulation via the master process. The details of the parallel scheme can be referred to [49].

2) Cloud-Based Parallel LS Framework: As a large IaaS service provider, Amazon EC2 provides a resizable compute capacity. We mitigate the parallel computing framework to a 16-core virtualized cluster of Amazon EC2 as shown in Fig. 3. Through the IaaS Web portal (located in the abstracted service layer), the simulation programs can be easily hosted and run on the virtualized resources, which are managed by the Amazon EC2’s virtual machine management schemes (located in the firmware layer). The parallel programs are scheduled by Amazon EC2’s underlying task scheduling schemes (located in the dynamical scheduling layer) on the physical resources. All the data communications are encrypted by the RSA to ensure the data security.

We use the same experimental setup with [49]. The total execution time of the simulation on a local PC is 1800 s. We evaluate the total computation time under different number of slave processes created on the cloud. The result is shown in Fig. 4. The total execution time significantly reduces with the increase in the slave processes. Fig. 5 reports the convergence curves of the parallel-DE on the cloud. It shows the parallel-DE finds a good enough solution only after nine iterations.

C. Data-Intensive Application Demonstration: Cloud-Enabled Pattern Discovery-Based Power System Dynamic Security Assessment

This section demonstrates how to utilize the cloud-enabled development tool to develop a dynamic security assessment (DSA) application and deploy it into the Google Application Engine (GAE) as an online service.

1) Introduction of GAE: GAE [70], [71] is a cloud service which provides an integrated framework (located in the abstracted service layer) for users to develop scalable Web applications directly on Google’s infrastructure, rather than pain access to the physical hardware. Users only need to focus on the logic of the applications while GAE automatically handles the common issues in other layers, such as load balancing, server environment set up and maintaining. Fig. 6 shows the GAE architecture.

Basically, every request is handled firstly by the App frontends, and then the App frontends read the configuration information of the application and dispatch it to an application server according to a certain load balancing mechanism. All the communications are encrypted by the RSA framework. The application then starts and runs in the application server. In this paper, the simulation program is developed by Java and
the development environment is Eclipse 3.7, which integrates the GAE plug-ins.

2) Introduction of PD-Based DSA: References [72] and [73] applied a pattern discovery algorithm on the DSA. PD trains n-dimension data tuples and partitions each dimension into several segments. By combining the partitions segments of each dimension, multiple hyper rectangles are identified in the data space, which are called events. Some events might be recursively partitioned. The hyper rectangles where the numbers of data points they contained are statistical meaningful are called patterns. Each pattern is labeled as secure/insecure according the percentage of the secure training data points it contains. After discovering the patterns, each online operating point (OP) can be assessed to be secure or insecure. The further details of the PD-based DSA can be found in [72] and [73].

3) Cloud-Based Framework for PD-Driven DSA: Although, PD can provide high accurate assessment results [72], one performance bottleneck of PD is that when the data dimension is high, the recursive process might be very time-consuming. Imaging a 20-D data space where each dimension will be divided into two segments during the partition, $2^{20}$ events will be formed. PD then needs to check the $2^{20}$ events one by one to see whether it should be further portioned. If so, then repeats the partition process on the subspace covered by that event. If there are too many events need to be recursive partitioned, the total execution time will become very large. Therefore, it is necessary to design some parallel mechanism to improve the performance of PD. On the other hand, it is desirable to publish the program as an online service, so that engineers can access it when they want to assess the OPs.

The parallel strategy here is to share the event checking task with multiple processes. The master process starts multiple processes to conduct the event checking and recursive partition task. Each process checks part of the events and send the result back to master. The parallel framework is shown in Fig. 7.

The training data are stored in the data store of GAE, which is a persistence database storing the data shared by all application instances. The master application reads the training data to do partition, and forms the event list. Then the master stores the in-formation of the events into data store, and sends n HTTP requests to slave application, where each request encapsulates start and end indexes of the events to be processed.

For every HTTP request, GAE starts and manages an instance of the slave application automatically. That instance reads the training data from data store, extracts the event indexes from the HTTP request, and do the checking and recursive partition. Finally, each slave updates the event information in data store and sends HTTP response back to the master.

4) Simulation Result: First, a transient stability database is artificially generated as the training database. Totally 6000 different OPs are simulated and the time-domain simulations are performed on the OPs under various disturbances. According to the time-domain simulation result, the OPs are labeled as secure or insecure. Then, 20 critical features are selected by employing the RELIEF algorithm [74]. Afterward, the PD algorithm is employed to discover patterns in the 20-D data space. We repeat the experiments 15 times on GAE by increasing the number of the parallel slave instances, and observe the computation time and network overhead as shown in Fig. 8.

It can be seen that the computation time is dramatically reduced along with the increase in parallel jobs, while there exist a little overhead, about 6–10 s. Those overheads are mainly incurred by starting the instances, TCP communications, and database access.

The performance of the cloud-based PD and single PC-based PD is compared by setting the number of the slave instance of the cloud-enabled PD to be 10 and varying the number of dimensions from 10 to 20. The single PC-based PD is performed on a 64-bit, Dual CPU DELL PC, with Windows 7 operation system. The computation time comparison and the corresponding network overheads of the cloud-based PD are shown in Fig. 9. It can be seen that when the data dimension increases, cloud-based PD shows significantly better performance than the single PC-based PD.

When the data dimension is 2 or 3, the pattern discovery can be visualized. Fig. 10 shows the patterns discovered with three selected critical features. Each rectangle represents a pattern. The technique guide of developing and publishing the Web applications through GAE can be found in [71].

D. Collaborative Application Demonstration: Cloud-Enabled DLC Framework

1) Application Overview: In this application, we propose a dispatch application of the large scale of thermostatically
controlled loads (TCLs). It is assumed that the large-scale TCLs are grouped and managed by multiple TCL aggregators. The aggregators participate in the DLC program launched by the utility, where the utility needs to dispatch the TCLs to achieve a specific objective (e.g., peak load shaving). The dispatch schematic of the TCLs is shown in Fig. 11. It is a collaboratively hierarchical dispatch model.

1) The aggregators submit the LS bids to utility.
2) The utility solves the upper-level day-ahead dispatch model according to the specific objective, and then sends the LS instructions to the aggregators.
3) After receiving the LS instructions, the TCL aggregators solve the lower-level day-ahead operational planning model and send the dispatch deviations back to the utility.
4) Based on the received dispatch deviations, the utility adjusts the LS instructions and sends the new instructions to the aggregators.
5) Repeat steps 4) and 5) until the predefined termination conditions are satisfied.

2) Utility Side (Upper-Level Dispatch Model): After receiving the bids, the utility solves the dispatch model to make the dispatch plan by assigning instructed LS amount to each aggregator. The day-ahead dispatch model of the utility aims to minimize the sum of the total LS cost and the penalty cost of the dispatch deviation between the upper and lower levels

\[
\min \left( \sum_{t=1}^{T} \sum_{a=1}^{A} \text{pr}^{\text{bid}}_{a}(t) \cdot D_{a}(t) + \sum_{t=1}^{T} \sum_{a=1}^{A} \alpha \cdot \text{dev}_{a}(t) \right)
\]

\[
\text{dev}_{a}(t) = |SL_{a}(t) - D_{a}(t)|
\]

s.t.

\[
0 \leq D_{a}(t) \leq C_{a}^{\text{TOTAL}}(t)
\]

\[
\sum_{a=1}^{A} D_{a}(t) \geq P_{\text{sys}}(t)
\]

where \(t\) and \(T\) are the index and number of the dispatch time intervals; \(a\) and \(A\) are the index and number of the TCL aggregators; \(\text{pr}^{\text{bid}}_{a}(t)\) is the bidding price of aggregator \(a\) at time \(t\) ($);
$D_a(t)$ is the instructed LS amount for aggregator $a$ at time $t$ (MW); $SL_a(t)$ is the scheduled LS amount of aggregator $a$ at time $t$ (MW); $\alpha$ is the penalty cost factor of the dispatch deviation between the two levels; $CP_a^{total}(t)$ is the total controllable TCL capacity in the bid of aggregator $a$ at time $t$ (MW), which can be determined by a certain bidding strategy; and $P^{sys}_t$ is the totally required LS of the system at time $t$ (MW).

3) Aggregator Side (Lower-Level Dispatch Model): The objective of the operational planning of the TCL aggregators is to maximize the total profit in the power market.

At time $t$, the scheduled total shed air conditioner load (ACL) of aggregator $a$ is

$$SL_a(t) = \sum_{g=1}^{G_a} S^g_a(t) \cdot CP^g_a$$

(5)

where $G_a$ is the number of the TCL groups of aggregator $a$; $S^g_a(t)$ is the state of the gth TCL group at $t$ (1-ON and 0-OFF); $CP^g_a$ is the capacity of the gth TCL group of aggregator $a$ (MW), which is calculated by the sum of the rated power of each TCL in the group. The income of the aggregator $a$ from the power market is represented as

$$\text{Profit}(a) = \sum_{t=1}^{T} \left[ p^\text{clear}_a(t) \cdot D_a(t) \cdot \Delta t \right] \left\{ SL_a(t) \geq D_a(t) \right\}$$

$$\sum_{t=1}^{T} \left[ (p^\text{clear}_a(t) \cdot SL_a(t) - \beta \cdot \gamma_a(t)) \cdot \Delta t \right] \left\{ SL_a(t) < D_a(t) \right\}$$

$$\gamma_a(t) = \max \{ 0, (dev_a(t) - dev_{max}) \}$$

(6)

(7)

where $\Delta t$ is the time interval duration; $p^\text{clear}_a(t)$ is the electricity clearance price in the day-ahead market at $t$ ($\$$); and $\beta$ is the penalty cost factor of the lower-level dispatch deviation.

The LS cost of the aggregator $a$ is calculated as

$$\text{cost}(a) = \sum_{t=1}^{T} p^\text{retail}_a(t) \cdot SL_a(t) \cdot \Delta t$$

(8)

where $p^\text{retail}_a(t)$ is the retailing price of the aggregator $a$ to its customers ($\$$). The aggregators solve model (9) to schedule the control actions of the ACL groups to maximize its total revenue

$$\text{Revenue}(a) = \text{Income}(a) - \text{cost}(a)$$

(9)

subject to the following.

1) TCL group state constraint

$$S^g_a(t) \in (0, 1) \forall a = A; g = 1; G_a, t = 1; T.$$  

(10)

2) Minimum online time constraint

$$\tau_{a,g}(t) \geq \tau_{\min}$$

$$\tau_{a,g}(t) = \left( \tau_{a,g}(t-1) + S^g_a(t) \cdot \Delta t \right) \cdot S^g_a(t)$$

(11)

(12)

where $\tau_{\min}$ is the minimal required TCL online time (hour); $\tau_{a,g}(t)$ is the accumulated online time of the gth TCL group of aggregator $a$ at time $t$.

4) Cloud-Based DLC Architecture: The proposed DLC framework involves a big data aggregation process, where the TCL aggregators need to collect the state data of the large-scale TCLs managed by them to do the lower-level dispatch. It also involves a typical distributed optimization process, where the utility and the aggregators communicate with each other to converge to the final solution iteratively. There are some basic communication requirements in this collaborative process. First, the communication architecture should be scalable, so as to support a large number of TCLs and TCL aggregators to participate in the DLC program; second, the communication system should be robust enough to ensure the reliable operation of the distribution network; third, the transferred data should be kept secure and confidential during communication; and lastly, the communication must be efficient.

In the current centralized system structure, the DLC process will be centrally performed in the EMS of the utility and the EMSs of the aggregators, respectively. In the lower-level dispatch, the utility establishes communications with the aggregators, and iteratively exchange information with them until the algorithm converges. In the lower-level dispatch, each aggregator communicates with the TCL groups, and solves the lower-level planning model. This can be depicted in Fig. 12.

Such centralized architecture has many limitations. First, it is a host address-centric architecture, meaning that the EMS of the utility and the aggregators need to know the host address of each other to communications. Such host address centric communication pattern is vulnerable to cyberattacker, because the cyberattackers can easily launch the DDoS attack to break down the EMS server; second, although the centralized architecture is suitable for small scale networks, it is very difficult to be scaled to cater for the large deployment of TCLs due to the bandwidth bottleneck and the computing capacity limit of the EMS.
Based on the cloud-based information infrastructure, the cloud-enabled DLC architecture is proposed to overcome above drawbacks, shown in Fig. 13. In the cloud-based architecture, the roles of the utility EMS and aggregators are weak, and the cloud acts as the collaboration intermediary. The EMSs of the TCL aggregators and the EMS of the utility interact with the cloud services. First, the sensor and actuator network layer collects the state data of the managed TCLs, and delivers it to the TCL aggregators through the data query service, which is located in the abstracted service layer. Based on the collected TCL data, the aggregators make the bids and send the bid information to the lower-level dispatch service. The lower-level dispatch service notifies the utility EMS about the bid information and then launches the distributed optimization process. In each iteration, the upper-level dispatch service notifies the lower-level dispatch service about the upper-level dispatch results, and the latter solves the lower-level dispatch model, and sends the updated lower-level dispatch results to the upper-level dispatch service. The upper-level dispatch service then updates the upper-level dispatch results. When the optimization process converges, the Web services notify the aggregator and the utility about the final dispatch results. Both of the lower-level and upper-level dispatch services are located in the abstracted service layer. The cloud-based DLC architecture is distinguished with the centralized DLC architecture in the following aspects.

1) **Service-Oriented Computing:** In the cloud-based architecture, the aggregator agents communicate with the utility and the TCL database through the abstracted service layer without knowing their host addresses.

2) **Data-Centric Distributed Data Management:** In the centralized DLC model, the TCL information is stored centralized in the database of the aggregator EMSs. In the cloud-based architecture, all the raw physical data (including demand side data and power network data), which is called observations [29], is stored distributed storage network. Users can define events to query the data. Events are referred to certain predefined constellations of observations. In our application,
model, and the AMPL solver [78] is applied on the upper-
\(\alpha\) parameter deviations. Groups to follow the load shed instructions, with some minor solid and dotted lines represent the instructed shed load and \(\alpha\) dispatch deviation decreases. However, with larger iteration times are needed to converge. The choice of \(\alpha\) is a compromise between dispatch deviation and execution time.

IV. CYBERSECURITY CONSIDERATIONS

Since the modern power systems have been becoming then complex cyber-physical systems, the cybersecurity issues of the power grids have been drawn increasingly attentions in recent years. As discussed below, with the support of some newly emerged cyberattack defend technologies, the cloud-based information infrastructure is capable of securing the cybersecurity collaborative process to ensure the reliable DLC operation.

Here we consider four kinds of cyberattacks as below, which have been proven to be the non-neglectful threatens for the modern industrial systems [79].

A. Compromised Key Attach and Eavesdropping Attack

By using the compromised key, the attacker can gain access to the secure communication between the load aggregator and the cloud to interpret the private information such as the price and the dispatch capacity. Also, since such private data will be decrypted in the cloud before the computing, the attacker could eavesdrop the decrypted data.

As for these two kinds of attacks, there are some new technologies can be integrated into the cloud-based information infrastructure to secure the information communications. One possible option is to utilize the SMC technology, which allows the communication ends to operate the encrypted data without decrypting it. For example, in the previous collaborative DLC application scenario, the utility can do the calculations based on the encrypted price and capacity data directly. By this way, there is no need to transfer the private key from the load aggregator to the utility agent, and the private data security can be significantly enhanced. The other possible solution is to adopt the “crypto cloud computing structure” [80], which is based on the quantum direct key system [81]. In the crypto cloud computing structure, each entity encrypts data using his/her own private key. All elements in the system have their own keys, and all the events occurred in the cloud environment are also assigned a unique key. For instance, in the collaborative DLC application, the involved Web services and all the DLC events (TCL data query, upper-level dispatch, lower-level dispatch, etc.) are assigned unique keys, which are hard to compromise by the attackers. In this way, the security and credibility of the exchanged information can be guaranteed.

B. Denial-of-Service Attack

The attacker can inject a huge amount of data to the communication channel to overload the server. The denial-of-service (DoS) attack could happen in different levels of the cloud environment. It could inject the junk data to consume the bandwidth resources (the bandwidth attack); it could take advantage of the lacuna of the network protocols to overload the target server (the protocol attack); or it could sends a large number of HTTP requests to attack the Web applications (the application attack). Taking the collaborative DLC application as an example, there is an iterative convergence process for the DLC dispatch, where the information is exchanged periodically between the servers where the DLC Web services are hosted. Any kind of above three DoS attacks will significantly block or delay the iterative DLC process.

The DoS attack in the cloud environment could be well addressed by many cloud-based solutions. For example, one option is to use the cooperative intrusion detection
A. Computational Load Scheduling

The cloud-enabled computational load scheduling algorithms for the future power system problems need to be studied. Different power grid problems often have different computing response time requirements. For example, a long-term network planning task has low accomplishment time requirement; an operational planning problem often needs to be resolved in several minutes to several hours; a real-time dispatch and control task may have to be accomplished in several seconds. Furthermore, different applications may need to access different power resources, which would be located in different locations and have different access authorities. Although the cloud-based information infrastructure can provide elastic and scalable computing capabilities to serve various power grid applications, the “power grid-aware” computational load scheduling algorithms need to be studied, which need to consider not only the performance of the computational resource network, but also the characteristics and performance requirements of the power grid applications.

B. Distributed Data Management

Different power applications may have different scale data requirements. For example, a distribution system planning application will only use the local system data to do optimization, and the control objectives are the local resources (feeders, transformers, substations, etc.); a wide-area control application may involve the large-scale data and control the resources distributed in the wide area. Therefore, the “smart” data aggregation and replication algorithms in the cloud-based information infrastructure should be studied to aware of the power grid features (such as the topology of the power grid), so as to achieve the optimal data retrieving, routing, and storage performance.

C. Hardware and Software Technique Obstacles

Although there have been many emerging technologies to support the implementation of the cloud-based information infrastructure, there are still some technical obstacles. On one hand, some promising technologies are still not mature enough to be adapted to wide range of power applications. For instance, although the SMC technique has been implemented to a certain extent and has been applied on some simple electricity price clearance calculations [92], the current SMC technology is still limited to perform complex numerical calculations [45]. This will restrict its applications, at least in the current stage. On the other hand, the availability of some power devices to support specific functions are limited. For example, some current smart grid devices may not support public key operations, which may lead to some security threats.
D. Impact of the Operation of the Cloud Data Centre on Power Systems

With the development of the cloud computing, the cloud data centers have become the large power consumers. For example, Microsoft’s data center in Quincy, WA, USA, has 43600 m² of space and uses 4.8 km of chiller piping, and 965 km of electric wire. This data center totally consumes 48 MW power, equal to 4000 homes [93]. Such large power consuming will also in turn affect the power flow of the grid. Therefore, the cloud data centers can also be treated as a specific kind of controllable load, and the proper computational load scheduling among different data centers can alter their energy demand, and finally make contribution to the effective operation of the power grid. Although the study of this issue can be found in [94], it is still a less-examined topic in both of the industry and of the academic circle.

E. Other Limitations

Other limitations will also be considered in the practical deployment of the cloud-based information infrastructure for the power grids. One limitation is the security aspect. As a newly developed computing paradigm, although there have been many efforts made to enhance the security of the cloud environment, it security still has not been proved for the large industrial systems. Considering the number of the participants of the future’s power grid is very large and the coordinated work flow of different power tasks would be very complex, how to promote the effective operation of the grid while keeping the data security and integrity will become a major concern.

Also, the operation of the cloud platform heavily relies on the Internet. Although the efficiency of the Internet is kept on improving, in many cases the bandwidth limitations or unstable network conditions would limit its applications on solving the scientific and engineering problems. Consider the local resources (such as the local HPC facilities) have the advantage of easy to access and deploy, for specific types of the power applications, the proper choice should be made on running the power applications on local resources or the cloud side.

Another limitation lies in the application development complexity. The whole cloud-based information infrastructure would be supported by different cloud vendors, and the cloud working environment provided by different vendors might be quite different. Furthermore, in some cases the development of the cloud-based application would require the expert knowledge in the domain of computer science, which would be more complicated than developing the applications on the local computing resources. Therefore, the power users might feel difficult to adapt themselves to the cloud development environments.

To sum up, due to some security, confidentiality, and convenience considerations, not all power applications are suitable to be mitigated to the cloud side. To what extent the power applications need to be mitigated to the cloud is an issue which needs to be investigated by different parties in the practical deployment.

VI. Conclusion

This paper gives the discussion of constructing the cloud-based information infrastructure for the next-generation power grid. First, this paper analyzed the limitations of the current information infrastructure of the power grid. Then, this paper proposed a layered model of the cloud-based information infrastructure for the power grid. Different cyber-physical entities are decoupled and their roles are identified in the layered model. The implementation technologies of the proposed layered model are also discussed.

Followed by the proposed layered cloud-based information infrastructure model, this paper discusses the benefits of the cloud computing technology to four categories of power applications, respectively. Then, this paper gives three specific cloud-based power applications, for the demonstration purpose. First, as an demonstration of the compute-intensive application, a LS strategy is constructed based on the IaaS service provided by the Amazon EC2; second, as a demonstration of the data-intensive application, a PD-based DSA application based on the PaaS and SaaS services provided by the GAE is described; third, as a demonstration of the collaborative smart grid application, a cloud-based architecture for dispatching large scale TCLs is developed.

The cybersecurity issues related to the cloud-based information infrastructure and some challenges and the technique obstacles for applying cloud computing technology on the next-generation power grid are also discussed in this paper.

ACKNOWLEDGMENT

The authors would like to thank G. Liang for her valuable comments and discussions on the cybersecurity issues of the industrial systems.

REFERENCES


