A Fog-Based Internet of Energy Architecture for Transactive Energy Management Systems

Mohammad Hossein Yaghmaee Moghaddam, *Senior Member, IEEE*, and Alberto Leon-Garcia, *Fellow, IEEE*

***Abstract*—Internet of Energy (IoE) is a subset of the Internet of Things which covers all aspects of electrical energy systems and provides secure connectivity and interoperability between power grid and Internet. In this paper, we present a fog-based IoE architecture for transactive energy (TE) management systems. The proposed design consists of three different layers. In the first tier, home gateways are employed which collect customers energy consumption data and provide necessary interface between customers and power grid. In the second layer, there are some local fog nodes located at the network edge and provide services with low latency. From the TE system point of view, the fog node act as retail energy market server which provides energy services to the end users. In the third layer, cloud servers are utilized to provide permanent and reliable data storage and high computing power. The proposed architecture supports different communication protocols such as hypertext transfer protocol, constrained application protocol, and OpenADR. We calculate the required bandwidth and delay performance of both fog- and cloud-based models. We present an optimal day ahead energy consumption schedule and an intercustomer energy trading mechanism for exchanging energy between end users. The performance of the proposed architecture is evaluated in terms of different power grid and communication network metrics. Results confirm the superiority of the proposed architecture.**

***Index Terms*—Demand response (DR), fog/cloud computing, Internet of Energy (IoE), Internet of Things (IoT), optimization, smart grid.**

I. Introduction

THE ELECTRICITY system continues to transform from a centrally supplied and managed infrastructure to a highly distributed system with an increasing variety of distributed supply, storage, and responsive demand assets. Distributed energy resources (DERs) are local power sources as such as photovoltaics (PVs), small-scale wind turbines, combined heat and power, gas-fired micro turbines, backup generation, and energy storages that can be aggregated into the power grid to provide required customer energy. DER is typically installed by the consumers to supply all, or a portion of, their electricity load. Depending on the size and capacity of the DER technology used, it may also be able to supply excess power to the utility grid. Energy storage systems (ESSs) is another DER technology that covers different storage technologies such as mechanical storage, chemical storage, and thermal storage. ESS devices absorb power from the grid or local energy generators and return it later. When the power grid is under peak load conditions, DERs can be engaged to provide part of customer energy requirements which effectively reduce the overall load profile in the power system.

The optimal use of the DERs to meet both business and operational objectives of utility companies and customers is still is a great challenge to be addressed. Transactive energy (TE) which is a new methodology and is emerging to coordinate the operation of intelligent devices, can be utilized to address this challenge. According to the Gridwise Architecture Council [1], TE refers the use of economic or market-based constructs to manage the generation, consumption or flow of electric power within an electric power system. It provides joint market and control functionality. Transactive nodes (TNs) are physical points within an electrical connectivity map of the system where electrical energy flows through on them and are controlled based on economic incentives, in real time. TNs exchange information and make transactions in a decentralized way to ensure the scalability of the control system. Each TN utilizes two different signals called transactive incentive signal (TIS) and transactive feedback signal, which present the forecasted delivered cost of electric energy and the predicted aggregated power flow at a particular TN, respectively. Transactive signals are exchanged between the neighboring TNs, to balance supply and demand. Each TN responses to the system conditions through decisions regarding the behavior of local assets. TE has been widely used in many industrial and demonstration projects such as The Pacific Northwest Smart Grid Demonstration [2], [3], Pacific Northwest National Laboratory [4], and the other projects given in [5]–[10].

To balance total demand with the amount of supply, demand response (DR) programs can be utilized. For efficient DR program, the energy consumption and generation information should be tracked in real time. To achieve this goal, we need to deploy more remote sensing equipment capable of measuring, monitoring and communicating. As described in [11], the Internet of Things (IoT) can be used to furnish intelligent management of energy distribution and consumption in heterogeneous circumstances. In the recent years, by the growth

Manuscript received September 12, 2017; revised January 13, 2018; accepted February 10, 2018. Date of publication February 14, 2018; date of current version April 10, 2018. *(Corresponding author: Mohammad Hossein Yaghmaee Moghaddam.)*

M. H. Yaghmaee Moghaddam is with the Department of Computer Engineering, Ferdowsi University of Mashhad, Mashhad 91779, Iran (e-mail: hyaghmae@um.ac.ir).

A. Leon-Garcia is with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S3G4, Canada (e-mail: alberto.leongarcia@utoronto.ca).

Digital Object Identifier 10.1109/JIOT.2018.2805899

of IoT and digital technologies, smart grid has been becoming smarter than before. The future power grid needs to be implemented in a distributed topology that can dynamically absorb different energy sources. Internet of Energy (IoE) is a subset of the IoT which cover all aspects of electrical energy system involved in generation, transmission, distribution and consumption to provide secure connectivity and interoperability between power grid and Internet.

Handling the volume, variety, and velocity of IoT data needs many requirements including minimize latency, conserve network bandwidth, address security concerns, operate reliably, collect and secure data across a wide geographic area with different environmental conditions, and move data to the best place for processing. Traditional smart grid architectures do not meet all of these requirements. The ideal place to analyze most IoT data is near the devices that produce and act on that data. For this purpose, we need to develop an efficient, high-speed, fully integrated, reliable, and intelligent communication network. This can be met by utilizing the fog computing. Fog computing is a decentralized computing infrastructure in which data storage and computing power are distributed in efficient place between the end nodes and the core cloud servers.

Fog computing offers many advantages over cloud computing such as load balancing, more bandwidth utilization, minimal downtime, interconnectivity, enhanced quality of services (QoSs) and low latency. On the other hands, the combination of fog computing with IoT will bring many benefits, including local data processing, cache data management, local resource pooling, load balancing, and delay reduction. When IoT and fog computing are integrated together, the time-critical data are locally analyzed by the fog edge node devices resulting in lower latency. It should be noted that fog computing facilitates the interactions between things and make it possible that things collaborate very effectively with each other. By integration IoT and fog computing, the proposed model will be able to provide caching to enable information-centric ability in IoT.

In this paper, we present a fog-based IoE architecture for TE management systems. IoE is an aggregation of IoT and smart grid which provides an architecture to implement a real-time interface between the DERs and the customer devices such as buildings, offices, electrical devices, and electrical vehicles. To use the DERs efficiently, we propose an inter customer energy trading mechanism for exchange energy between end users. We also present an automated DR program based on TE systems. To transfer price signals and demand information between TNs, the open automated DR (OpenADR) standard is utilized. The OpenADR is a communications data model for automation of DR developed by Lawrence Berkley National Laboratory [12]. We present a convex optimization function to minimize both customer and utility company costs by the optimized scheduling of shiftable customer demand. By considering the growth of data that should be transferred, stored, processed, and movement toward big data we present a distributed fog-computing architecture which provides low latency requirement of energy management systems and prevents high bandwidth consumption. As the smart grid components such as power generators, microgrid, and distribution substations are naturally distributed, the proposed architecture fits in this nature. In the proposed architecture, the fog nodes communicate with the OpenADR gateway through a local area network (LAN) to collect customer energy consumption and generation information. Customers can access their detailed energy consumption data securely using their local fog node. Besides, fog nodes reduce the search and access time with their locality attribute.

One of our contribution in this paper is the use of fog computing capabilities in the smart grid networks. In many countries, smart power grid includes millions of users with thousands of kilometers of transmission lines which provides reliable electrical power to the customers. The modern power grid is inherently distributed and decentralized without any central controller. Instead of central energy generation, many DERs such as wind turbines and solar cell systems are utilized. The smart grid properties make it a prime target for fog computing applications. The huge volume of data produced by the smart grid sensors and the timing requirements of some control loops requires rapid, robust, and distributed sensor processing decision making that can only be provided by the fog computing. Note that the fog computing can be utilized for the other related tasks in the smart grid such as advanced metering infrastructure (AMI). In the AMI, smart meters are developed to transfer electricity information and monitor and manage the electricity consumption. When the number of smart meters increases, a huge amount of data is generated that is hard to process, analyze, and store. In this case, fog computing can be used to collect, compute, and store smart meter data which reduces latency, increases privacy and locality for smart grids. Knowing that IoT can be utilized for different applications of the smart grid and various area of energy production, this paper is an aggregation of IoT and smart grid which provides an architecture to implement a real-time interface between the DERs and the customer.

The contributions of this paper are summarized as follows.

1) Presenting a multitier communication architecture for efficient energy management in the IoE systems.
2) Proposing an intercustomer price function for direct energy trading between customers based on the TE concepts.
3) Presenting a TE-based DR program with a convex optimization function for minimizing both customer and utility company costs.
4) Providing a platform for customers data collection so that all customers can monitor their instantaneous power consumptions, the maximum, the minimum, and the average consumption of the community and their position in the community in term of power consumption. This information helps customers to take intelligent decision for their energy consumption.
5) Investigating the effect of communication channel on the bandwidth and delay performance of the fog and cloud-based DR programs.
6) Implementing and performance evaluating of the proposed architecture on a real testbed.

The rest of this paper is organized as follows. Section II explains some related work. Section III presents the proposed architecture consist of the system model, customer model, price function, power consumption and PV generation predictor, day ahead scheduling, bandwidth requirements, and delay performance of optimization program. Section IV presents the implementation and simulation results that confirm the superior performance of the proposed model. Finally, Section V concludes this paper.

II. Related Work

Fog computing extends cloud computing and services to the edge of the network where data is created and acted upon. Similar to cloud, fog provides data, compute, storage, and application services to end users. Fog computing is a term created by Cisco that it is motivation lies in a series of real scenarios, such as software defined networks, smart grid, smart traffic lights, connected vehicles, decentralized smart building control, wireless sensors, IoT, and cyber-physical systems. It improves the delay and bandwidth efficiency by reducing the amount of data transported to the core cloud servers.

As mentioned in [13], fog computing model relies on the assumption that computer tasks can be performed by nodes placed at the edge of a network. It provides so many advantages for developers. Fog can be distinguished from cloud by its proximity to end-users. It supports mobility by its dense geographical distribution, low latency, location awareness, and improves QoS and real-time applications. Typical examples of fog computing include industrial automation, transportation, and networks of sensors and actuators. The fog paradigm is well positioned for real time big data analytics, supports densely distributed data collection points, and provides advantages in entertainment, advertising, personal computing, and other applications [14].

While implementing applications for edge computing, we face increasing number of sensor devices and surging of cloud models that leads us to Internet of everything paradigm which refers to an ecosystem of edge devices that share their limited resources. We expect that fog computing's integration with Internet of everything bring up a number of new applications [15]. For example, smart cities applications are emerging to implement fog computing. As mentioned in [16], smart city vision brings emerging heterogeneous communication technologies such as fog computing together to substantially reduce the latency and energy consumption of the Internet of everything devices running various applications. The key feature that distinguishes the fog computing paradigm for smart cities is that it spreads communication and computing resources over the wired/wireless access network to provide resource augmentation for resource- and energy-limited wired/wireless (possibly mobile) things. Fog computing allows to deliver the right data at the right time to people on any device. The model for data analytics involves a centralized data warehouse manual data manipulation and investigation, with the vast amount of data of Internet of everything pouring. The smart grid, fog computing and the big data are three modern and important paradigms.

Fog computing has been utilized for different applications of the smart grid. Naranjo *et al.* [17] stressed the general benefits of using big data to design and support smart grid applications on the fog computing platforms. In [18], by presenting a mathematical model the performance of fog computing and cloud computing has been investigated. The tradeoff between power consumption and delay in a cloud-fog computing system has been studied. Results confirm that fog computing has a better delay and bandwidth performance than cloud computing. In [19], the use of a combination of the fog computing and microgrids with renewable energy sources, and local weather forecasting for reducing the energy consumption of IoT applications has been investigated. In [20] with the knowledge that by increasing the number of smart meters, the centralized information processing architecture will no longer be sustainable, a fog-based data storage-and-processing solution for improving the existing smart meter infrastructure has been proposed.

IoT can be utilized for various applications of the smart grid including distributed power plant monitoring, power generation and consumption prediction, power consumption monitoring, energy storage monitoring, smart meter, electric vehicle charging, power demand side management, and various area of energy production. In [21], by combining the constrained application protocol (CoAP) with the ubiquitous ID architecture, a new framework has been presented that can be used to implement different IoT applications over existing embedded systems such as usual consumer appliances. In [22] using the IEEE802.15.4 and ZigBee sensor network, a new smart home energy management system has been designed and implemented which divides and assigns various home network tasks to appropriate components. Langhammer and Kays [23] proposed a new methodology for the evaluation of wireless smart homes and home automation networks. Different wireless technologies used to interconnect devices in a smart home environment have been investigated and evaluated in realistic indoor scenarios. In [24], the concepts of IoT-grid which uses the befits of IoT in smart grid is introduced and then the communication aspect of IoT-grid is evaluated. The evaluation results confirm that processing delays of IoT devices have large impact on IoT-grid. To fix this problem, a mechanism based on sending burst commands with scheduled responses has been proposed. Spanò *et al.* [25] presented an architecture for customer domain part of the smart grid network. The proposed architecture is based on IoT platform which can host different smart home applications. Minoli *et al.* [26] explained different applications of IoT in industry and then reviews some of the existing opportunities and challenges of the IoT in the smart building area. Brundu *et al.* [27] presented and deployed an IoT software infrastructure for energy management in smart cities which enables the interoperability of energy profiles with environmental data from building and power grid sensors.

III. System Model

A. System Model

Fig. 1 shows the proposed design which introduces an IoE architecture based on fog computing for TE management. As

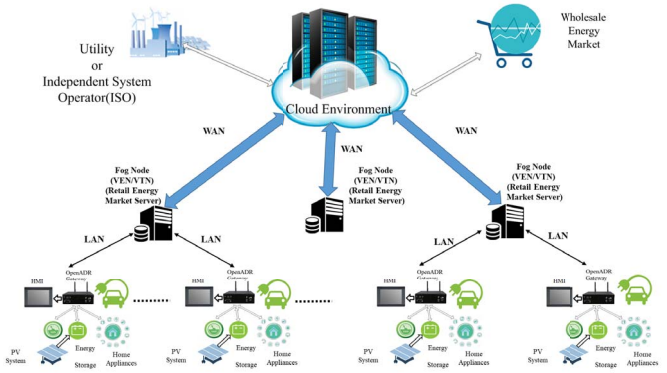


Fig. 1. Proposed fog-based IoE architecture for TE management.

it can be seen in this figure, the proposed design consists of three different layers.

In the first layer, home gateways are utilized which collect customers energy consumption information and the other related data and send data to the network servers using the OpenADR protocol. The OpenADR defines a client/server communication model which named server as virtual top node (VTN) and client as virtual end node (VEN). Nodes are organized in a tree and classified to VTN and VEN. Some nodes, such as like aggregators could have both roles of VEN and VTN at the same time [28]. A human–machine interface (HMI) screen is employed as the interface between the customer and the power grid. The communication between home gateways and home appliances can be done by utilizing traditional wired or wireless technologies such as IEEE 802.11 (WLAN) [29], IEEE 802.15 (WPAN) [30], and IEEE 802.15.4 (LR-WPAN) [31]. In the proposed architecture, the home gateway acts as VEN which monitors customer power consumption information in real-time. It provides customers useful information such as the instantaneous amount of energy uses, the power consumption status relative to the average, maximum and minimum consumption of the community, the power consumption of each electric device, and the optimized day ahead scheduling of customer appliances. This information helps customers to take intelligent decision for their energy consumption. According to the study given in [32], this information can affect customer behavior to reduce power consumptions.

In the second layer of the proposed architecture, there are some local fog nodes located at the edge of the communication network and provide services with low latency. Fog node is considered as a physical that implements fog computing and provides resources for services at the edge of the network. As the fog nodes are located at the network edges and are close to the customers, delay performance is improved. In the proposed architecture, the fog nodes have CPU and storage capacities, which would be used to process the customer data as well as external requests. The main goal of fog nodes in the proposed architecture is to improve efficiency and reduce the volume of data transferred to the cloud server for processing, analysis, and storage. Fog nodes act as VEN/VTNs and communicate with the home gateway through a LAN to collect data. To reduce stored data in the cloud servers, data aggregation protocols are employed at the network edge. Customers can access their detailed energy consumption information securely using their local fog nodes. Besides, fog nodes reduce the search and access time with its locality attribute. From the TE system point of view, the fog nodes act as retail energy market server which provides energy services to the end-users. Energy retail markets provide the interface between retailers and their customers. Based on the amount of existing sale energy and the grid energy price, the retail energy market server determines the intercustomer energy price and broadcasts TIS signal to all customers who have a lack of energy.

In the third layer of the proposed architecture, cloud servers are employed to provide permanent and reliable data storage and high computing power. The wholesale energy market and the utility or independent system operators are in connection with the cloud servers. The cloud servers get the grid price from the market and broadcast it to all retail energy market servers located at the network edge. Note that the communication between fog nodes and the cloud servers is done through the wide area networks (WANs) such as Internet.

B. Customer Model

As it can be seen in the bottom of Fig. 1, each customer's home equipped with some electric appliances, a plug-in electric vehicle (PEV) a PV system and an energy storage (battery). Customers first try to meet their own required energy and store the excess energy into their batteries. However, extra customers' energy can be shared amongst the other neighbors at the peak load hours. The home gateway provides interactions between customers and the power grid which allows effective operation and control of the system from the customer's end. The home gateway gets system parameters from the users and informs customers of the instantaneous electricity prices, their consumptions, their total costs, their energy consumption history, and the other useful information through the HMI screen. Each appliance connects to the home gateway through a home area network protocol. Privacy and data security are the main concerns of cloud-based applications in the smart grid. As cloud servers are used in a shared environment such as Internet; therefore, they suffer from lack of privacy. To solve this problem, the fog computing can be employed. The gathered data from the customers can be categorized into two different categories, private and public. Power consumptions of each appliance, type and the number of appliances in each customer's home are some examples of private data which should be stored in secure and safe data base. Total energy consumptions and the amount of renewable energy generation are not highly private data and could be considered as public data. Fog computing can provide data privacy by separating the public and private data. The private data is stored in the local fog nodes while the public data is forwarded to the cloud server for further processing. In this architecture, customers can achieve their detailed information with low latency.

The PV system generates electricity and stores it for further consumption in the energy storage. The home gateway also monitors the battery state of charge (SoC). Suppose N and

$\mathbb{N}$ represent the number of customers and the customer's set, respectively. Let $\mathbb{A}_i$ denote the set of appliances a of customer $i \in \mathbb{N}$. Suppose the time is divided into a series of time slots of length $\triangle t$. Let $l^t_{\text{PEV},i}$ and $l^t_{a,i}$ denote the power consumption of PEV and appliances a of customer i at time slot t, respectively. Let $l^t_i = l^t_{\text{PEV},i} + \sum_{a\in A_i} l^t_{a,i}$ denotes the energy consumption of customer i at time slot t. Suppose b^t_i and B^F_i represent the amount of energy stored in the battery of customer i at time t and the nominal battery capacity of customer i, respectively. Due to the limited capacity of local storages, b^t_i should always be less than B^F_i $(0 \le b^t_i \le B^F_i)$.

Let g^t_i and y^t_i denote the amount of energy generation and battery consumption of customer i at time t, respectively. As the battery is charged by solar energy (g^t_i) and is discharged by local consumption (y^t_i), we always have

$$b^t_i = b^{t-1}_i + g^t_i - y^t_i. \tag{1}$$

C. Price Function

At the beginning of each time slot t, the power grid determines the grid energy price, pr^t_G, and forwards it to each fog nodes in the system. Also, the fog nodes collect the customer energy consumption data and their battery status. The utility company informs customers about peak and nonpeak load times through the local fog nodes. So, at the beginning of each time slot t, fog nodes get the status of the power grid in terms of peak or nonpeak load conditions and the energy price. According to the status of total power consumption, and the battery state of each customer i at time t, the following rules are applied.

1) The stored energy in the battery of each customer (b^t_i) is always consumed at the peak load times. All required energy at the nonpeak load times is bought from the power grid with price pr^t_G.
2) At the peak load times, all (when $l^t_i \le b^t_i$) or part of customer consumption (when $l^t_i > b^t_i$) is supplied from the local battery.
3) When the power grid is at the peak load times, depends on l^t_i and b^t_i, each customer i is *energy buyer* or *energy seller*. When $l^t_i > b^t_i$, customer i is *energy buyer*, otherwise, the customer is *energy seller*.

Suppose $\mathbb{N}_B$ and $\mathbb{N}_S$ represent the sets of *energy buyers* and *energy sellers*, respectively. Note that $\mathbb{N}_B \cup \mathbb{N}_S = \mathbb{N}$, where $\mathbb{N}$ represents the customers set. Let $\mathbb{T}^p$ and $\mathbb{T}^{\text{np}}$, represent the set of peak load and nonpeak load times, respectively. Note that $\mathbb{T}^p \cup \mathbb{T}^{\text{np}} = \mathbb{T}$, where $\mathbb{T}$ represents the time set. Since the energy consumption and energy generation information of all customers is instantly available, there is a potential energy deal between *energy buyers* and *energy sellers*. In the proposed architecture, using the benefits of TE systems, the *energy buyers* prefer to buy energy from the *energy sellers* rather than buying from the power grid which is always more expensive at the peak load times. At the beginning of each time slot, the *energy sellers* send their estimation of extra energy for sale to the retail energy market server located on the fog node. Furthermore, the *energy buyers* also send their required energy to the server. At each time slot $t \in \mathbb{T}^p$, the total energy required by *energy buyers*, E^t_B, is equal to $E^t_B = \sum_{i\in\mathbb{N}_B} l^t_i - b^t_i$. Similarly, the total energy sales, E^t_S, is computed as $E^t_S = \sum_{i\in\mathbb{N}_S} b^t_i - l^t_i$. Note that at nonpeak time $t \in \mathbb{T}^{\text{np}}$, $E^t_B = \sum_{i\in\mathbb{N}_B} l^t_i$, and $E^t_S = 0$. We assume that the retail energy market server determines the energy price and sends the TIS signal to the all *energy buyers*. Let pr^t_{TE} denotes the sale energy price in the TE system at time slot $t \in \mathbb{T}$ which is determined by the retail energy market server on behalf the *energy seller*s. Based on the supply law, when there is not any energy for sale by customers, the energy price is high and equal to the power grid energy price. However, the customer sale price decreases by increasing the available sale energy. The following price function is proposed to determine the energy sale price pr^t_{TE} at each time $t \in \mathbb{T}$:

$$\text{pr}^t_{\text{TE}} = \begin{cases} \text{pr}_{\min}, & \text{if } E^t_B \le E^t_S \\ \frac{\text{pr}^t_G - \text{pr}_{\min}}{E^t_B}\left(E^t_B - E^t_S\right) + \text{pr}_{\min}, & \text{else} \end{cases} \tag{2}$$

where $\text{pr}_{\min}$ is the minimum value of sale energy price. The following relationship is always established:

$$\text{pr}_{\min} \le \text{pr}^t_{\text{TE}} \le \text{pr}^t_G, \ t \in \mathbb{T}^p. \tag{3}$$

Suppose C^t_B, represents the total energy cost of the *energy buyers* at time $t \in \mathbb{T}$. C^t_B is calculated as follows:

$$C^t_B = \begin{cases} E^t_B \ \text{pr}^t_{\text{TE}}, & \text{if } E^t_B \le E^t_S \\ E^t_S \ \text{pr}^t_{\text{TE}} + \left(E^t_B - E^t_S\right) \ \text{pr}^t_G, & \text{else.} \end{cases} \tag{4}$$

The total revenue of the *energy seller*s at each time slot $t \in \mathbb{T}$, R^t_S, is calculated as follows:

$$R^t_S = \begin{cases} E^t_B \ \text{pr}^t_{\text{TE}}, & \text{if } E^t_B \le E^t_S \\ E^t_S \ \text{pr}^t_{\text{TE}}, & \text{else.} \end{cases} \tag{5}$$

By substituting the price pr^t_{TE} given in (2) into (4) and (5), we have

$$C^t_B = \begin{cases} E^t_B \ \text{pr}_{\min}, & \text{if } E^t_B \le E^t_S \\ E^t_B \ \text{pr}^t_G - {E^t_S}^2 \ \frac{\text{pr}^t_G - \text{pr}_{\min}}{E^t_B}, & \text{else} \end{cases} \tag{6}$$

$$R^t_S = \begin{cases} E^t_B \ \text{pr}_{\min}, & \text{if } E^t_B \le E^t_S \\ E^t_S \ \text{pr}^t_G - {E^t_S}^2 \ \frac{\text{pr}^t_G - \text{pr}_{\min}}{E^t_B}, & \text{else.} \end{cases} \tag{7}$$

D. Power Consumption and PV Generation Predictor

The proposed architecture can find the optimized day ahead power consumption schedule. For this purpose, prediction techniques are utilized to predict the power consumption and PV generation. The most dominant prediction technique is time-series analysis which can be used to find repeating patterns in the historical data to forecast future values. Consequently, the scaling action is done in advance. Some of the techniques used for this purpose in the literature are moving average, autoregression, ARMA, exponential smoothing, and machine learning approaches. Among the different prediction algorithms, the normalized least mean square (NLMS) [33] predictor is the one providing the best tradeoff between complexity, accuracy, and responsiveness. We consider the PV generation and power consumption as a stochastic process $F(n)$. The NLMS predictor generates an estimation $\hat{F}(n+k)$ of the value $F(n+k)$ that the process F will assume k steps ahead. In other words, given a vector of

p observations, $\underline{F} = [F(n), F(n-1), \ldots, F(n-p+1)]$, the predicted value $\hat{F}$ is obtained by $\hat{F} = \psi(\underline{F})$ where the function ψ is called predictor. Based on the previous studies, the linear prediction class whenever the function ψ is linear, is the best suited for our aim. The problem then is to determine the impulse response $q(n)$ of the linear filter q such that

$$\hat{F}(n+k) = F(n) \otimes q(n) = \sum_{i=0}^{p-1} q(i)F(n-i). \tag{8}$$

The NLMS algorithm is based on an adaptive approach. It does not require prior knowledge of the autocorrelation structure of the stochastic sequence. The filter coefficients are time varying and are tuned on the basis of the feedback information carried by the error $\in (n)$. In the following, we denote the vector of filter coefficients at time n with $\underline{q}_n$. The values of $\underline{q}$ adapt dynamically in order to decrease the mean square error. Notice that $\mathcal{E}(n) = F(n+k) - \hat{F}(n+k)$. The NLMS algorithm operates as follows.

1) Initialize the coefficient $\underline{q}_0$.
2) For each new data, update the filter $q(n)$ according to the recursive equation

$$\underline{q}_{n+1} = \underline{q}_n + \mu \frac{\mathcal{E}(n)\underline{F}_n}{\|F_n\|^2}$$

where $\|F_n\|^2 = \underline{F}_n \underline{F}_n^T$ and μ is a constant called step size.

The historical data can be used to tune the filter parameters. The NMLS predictor needs the configuration of two parameters: 1) the order p and 2) the step size μ. These parameters should be set correctly so that the best performance with minimum error is obtained. In the case of the μ, it is relevant to note that one of the main advantage of using NLMS is that it is less sensitive to the step size with respect to other linear predictor.

E. Day Ahead Scheduling

In this section, we introduce the proposed day ahead scheduling mechanism. Optimizing the energy cost of the customers have been already known as an important issue. In [34] by using a day-ahead demand-side management mechanism, customers are interested in reducing their energy cost by producing or storing energy through their local DER rather than purchasing their energy needs from the power grid. A grid optimization problem using a noncooperative method and nonlinear programming approach have been formulated. Nguyen and Le [35] proposed a district energy management system to control the real-time energy consumption of the customers and minimize purchasing energy from the grid. A linear optimization problem is formulated which considers a microgrid with renewable energy sources, storage systems and electric vehicles. In [36], an optimization framework for energy management in a cooperative network of micro grids that seeks to achieve an efficient tradeoff between low operation cost and good energy service for customers has been proposed.

Suppose each customer has equipped with two types of appliances, shiftable and nonshiftable. Let $E^t_{\text{PEV},i}$ represents the available energy in battery of PEV of customer $i \in \mathbb{N}$ at time t, respectively. Suppose $E^D_{\text{PEV},i}$, $E^{\min}_{\text{PEV},i}$, $E^{\max}_{\text{PEV},i}$, and θ_c represent the desired charging energy, the minimum and the maximum battery capacity, and the cycle charging efficiency of a PEV of customer i, respectively. The following equation is always satisfied [37]:

$$\begin{aligned} &E^t_{\text{PEV},i} = E^{t-1}_{\text{PEV},i} + \theta_c . l^t_{\text{PEV},i} \\ &E^{\min}_{\text{PEV},i} < E^t_{\text{PEV},i} < E^{\max}_{\text{PEV},i} \\ &E^{t\_\text{end}}_{\text{PEV},i} = E^D_{\text{PEV},i} \end{aligned} \tag{9}$$

where t_end represents the ending charging time of the PEV. Let $\mathbb{A}^s_i$ and $\mathbb{A}^{\text{ns}}_i$ denote the set of shiftable and nonshiftable appliances of customer $i \in \mathbb{N}$. Note that $\mathbb{A}^s_i U \mathbb{A}^{\text{ns}}_i = \mathbb{A}_i$. Let $\overleftrightarrow{l}^t_i = \sum_{a \in \mathbb{A}^s_i} l^t_{a,i}$ and $\vec{l}^t_i = \sum_{a \in \mathbb{A}^{\text{ns}}_i} l^t_{a,i}$ represent the power consumption of the shiftable and nonshiftable appliances of customer i at time slot t, respectively. The following equation is always satisfied:

$$l^t_i = \overleftrightarrow{l}^t_i + \vec{l}^t_i + l^t_{\text{PEV},i}. \tag{10}$$

Let $l^{\min}_{a,i}$, $l^{\max}_{a,i}$, $[S\_\text{time}^a_i, E\_\text{time}^a_i]$, and $E_{a,i}$ represent the minimum power level, the maximum power level, the desired operation time (start time and end time), and the total energy needed for shiftable appliances $a \in \mathbb{A}^s_i$. Note that for each customer $i \in \mathbb{N}$, the total power consumption of shiftable appliances during a day, $\overleftrightarrow{L}_i$, is always fixed and calculated as $\overleftrightarrow{L}_i = \sum_{t=1}^{\mathbb{T}} \sum_{a \in \mathbb{A}^s_i} l^t_{a,i}$.

Without loss of generality, we suppose the amount of energy requested by the *energy buyers* is always more than the energy provided by the *energy sellers* ($E^t_B > E^t_S$). Note that $L^t = E^t_B - E^t_S$ is the total load on the power grid at time t. The quadratic cost function has been widely used to model the cost of energy provided by utility companies as: $C^t_U(L^t) = a_t {L^t}^2$, where a_t is the cost coefficient at time t which is determined by some elements, such as operating costs, facility construction, and ownership cost [38], [39]. Let C^t_T represents the total cost of both customers (for buying energy) and utility company (for providing energy). C^t_T is defined as follows:

$$C^t_T = C^t_B - R^t_S + C^t_U. \tag{11}$$

By setting the values of C^t_B, R^t_S, and C^t_U in (11), we have

$$C^t_T = \left(E^t_B - E^t_S\right)\text{pr}^t_G + a_t\left(E^t_B - E^t_S\right)^2. \tag{12}$$

The proposed cost function C^t_T given in (12) is a quadratic cost function. It has been proven that quadratic cost function always has an optimum solution if and only if the cost function is increasing and strictly convex. It can be seen that the proposed cost function is increasing and strictly convex. It means that

$$\begin{aligned} &C^t_T\left(\hat{L}^t\right) < C^t_T\left(\tilde{L}^t\right) \quad \forall\ \hat{L}^t < \tilde{L}^t \\ &C^t_T\left(\varepsilon\hat{L}^t + (1-\varepsilon)\hat{L}^t\right) < \varepsilon C^t_T\left(\tilde{L}^t\right) + (1-\varepsilon)C^t_T\left(\tilde{L}^t\right). \end{aligned} \tag{13}$$

With the same goals as previous work presented in [34]–[36] to minimize the total cost, the energy consumption of each customer i at time t, l^t_i, should be scheduled. The

following convex optimization problem is defined to minimize the total daily cost of both users and utility company:

$$\text{minimize} \sum_{t\in\mathbb{T}} C_T^t = \sum_{t\in\mathbb{T}^p} a_t \left(\sum_{i\in\mathbb{N}_B} \left(l_i^t - b_i^t\right) - \sum_{j\in\mathbb{N}_S} \left(b_j^t - l_j^t\right) \right)^2 + \text{pr}_G^t \left(\sum_{i\in\mathbb{N}_B} \left(l_i^t - b_i^t\right) - \sum_{j\in\mathbb{N}_S} \left(b_j^t - l_j^t\right) \right) + \sum_{t\in\mathbb{T}^{\text{np}}} a_t \left(\sum_{i\in\mathbb{N}} l_i^t \right)^2 + \text{pr}_G^t \left(\sum_{i\in\mathbb{N}} l_i^t \right) \quad (14)$$

subject to

$$l_i^t = \overleftrightarrow{l}_i^t + \vec{l}_i^t + l_{\text{PEV},i}^t,\ i \in \mathbb{N},\ t \in \mathbb{T} \quad (14.a)$$

$$\overleftrightarrow{l}_i^t = \sum_{a\in\mathbb{A}_i^s} l_{a,i}^t,\ i \in \mathbb{N},\ t \in \mathbb{T} \quad (14.b)$$

$$\vec{l}_i^t = \sum_{a\in\mathbb{A}_i^{\text{ns}}} l_{a,i}^t,\ i \in \mathbb{N},\ t \in \mathbb{T} \quad (14.c)$$

$$l_{a,i}^{\min} \le l_{a,i}^t \le l_{a,i}^{\max}, a \in \mathbb{A}_i^s,\ i \in \mathbb{N},\ t \in \mathbb{T} \quad (14.d)$$

$$\sum_{t=S\_\text{time}_i^a}^{E\_\text{time}_i^a} l_{a,i}^t = E_{a,i}, a \in \mathbb{A}_i^s, i \in \mathbb{N},\ t \in \mathbb{T} \quad (14.e)$$

$$\sum_{t=1}^{\mathbb{T}} \sum_{a\in\mathbb{A}_i^s} l_{a,i}^t = \overleftrightarrow{L}_i,\ i \in \mathbb{N}, t \in \mathbb{T} \quad (14.f)$$

$$b_i^t = b_i^{t-1} + g_i^t,\ i \in \mathbb{N},\ t \in \mathbb{T}^{\text{np}} \quad (14.g)$$

$$b_i^t = g_i^t,\ i \in \mathbb{N},\ t \in \mathbb{T}^p,\ t \neq 1 \quad (14.h)$$

$$b_i^t = b_i^{\acute{t}} + g_i^t,\ t = 1,\ \acute{t} = \text{last member of set } \mathbb{T}^{\text{np}} \quad (14.i)$$

$$E_{\text{PEV},i}^t = E_{\text{PEV},i}^{t-1} + \theta_c . l_{\text{PEV},i}^t,\ i \in \mathbb{N},\ t \in \mathbb{T} \quad (14.j)$$

$$E_{\text{PEV},i}^{\min} < E_{\text{PEV},i}^t < E_{\text{PEV},i}^{\max},\ i \in \mathbb{N},\ t \in \mathbb{T} \quad (14.k)$$

$$E_{\text{PEV},i}^{t\_\text{end}} = E_{\text{PEV},i}^D,\ i \in \mathbb{N},\ t \in \mathbb{T}. \quad (14.l)$$

Constraint (14.a) confirms that total load of each customer at any given time is composed of two values including shiftable and nonshiftable load. According to Constraint (14.b) and (14.c), the total shiftable and nonshiftable load is obtained by aggregating the power consumption of each appliance in the given appliance's set. Constraint (14.d) shows that the power consumption of each shiftable appliances and at each time t is between the minimum and the maximum power levels. Constraint (14.e) confirms that total energy consumption of each shiftable appliances during its desired operation is limited to a fixed value. Also, constraint (14.f) shows that the total energy consumption of all shiftable appliances in all-time intervals is limited to a fixed value. According to constraint (14.g), at the nonpeak load time, the generated energy by the local PV systems is stored in the battery and not consumed. At the peak load times, all generated energy by PV system is completely consumed by the owner or is sold to the other customers. According to constraint (14.i), at the first time interval in the peak load times, there is some initial stored energy from the previous time slot in the nonpeak load. Constraints (14.j) confirms that the amount of vehicle energy in the current time slot is equal to its previous value plus the amount of energy absorbed during charging in the current time slot. According to constraint (14.k), the PEV energy in each time slot is always between the maximum and the minimum battery capacity. Constraint (14.j) confirms that at the end of charging process the PEV energy is equal to the desired charging energy. The proposed quadratic and convex problem (14) can be solved by a well-known and highly efficient algorithm interior-point-convex [40]. As it has been proven in [41], the most important advantage of convex optimization is that the local optimal solution is the global optimal solution.

F. Bandwidth Analysis

At the beginning of each time slot t, the home gateway collects the power consumption of each customer appliances, $l_{a,i}^t$, and the battery status, b_i^t, and then transfers them to the VTN, regularly. Note that the OpenADR gateway needs to register customers and their appliances in the VTN which is done only once at the beginning of the operation. This can be done by sending the registration service (EiRegisterParty) message.

To evaluate the superiority of the proposed fog-based architecture, we consider two different models including fog- and cloud-based models. Fig. 2 shows the detail of message transferring between the OpenADR gateway and the VTN in both two models. After registering of VEN, both VEN and VTN send their reporting capabilities. Then VTN can send its request of periodically automated reading to the VEN. Then the VEN start regularly sending messages in specified intervals containing power consumption of the individual device using the report service (EiReport) message. VTN sends customers total consumption and costs in semi real-time to the VEN. Also, the minimum, maximum, and average power consumption in the community will be sent to let the customers know their power consumption in the community and change their behavior to reduce their consumptions. The proposed OpenADR gateway implementation is based on simple hypertext transfer protocol (HTTP) transport which is ideal for simple implementations and essentially represents a scaled down representational state transfer (REST) implementation.

At each time interval $\triangle t$, the OpenADR gateway collects power consumption information of each appliances $a \in \mathbb{A}_i$ and also the battery status and transfers them to the VTN located at the fog/cloud server. Suppose S_i represents the required message size for transferring power consumption information of customer i from the home gateway to the local fog node. S_i is calculated by the following equation:

$$S_i = H_{\text{Size}} + |\mathbb{A}_i| \left|l_{a,i}^t\right| + \left|b_i^t\right| \quad (15)$$

where $|\mathbb{A}_i|$, $|l_{a,i}^t|$, and $|b_i^t|$ are the cardinality of $\mathbb{A}_i$ and the size of $l_{a,i}^t$ and b_i^t in bits, respectively. H_{Size} is the header size of the underlying communication protocols including HTTP, transmission control protocol (TCP), Internet protocol (IP), and the network interface protocols in bits. VTN calculates the maximum, minimum, and the average power consumptions of

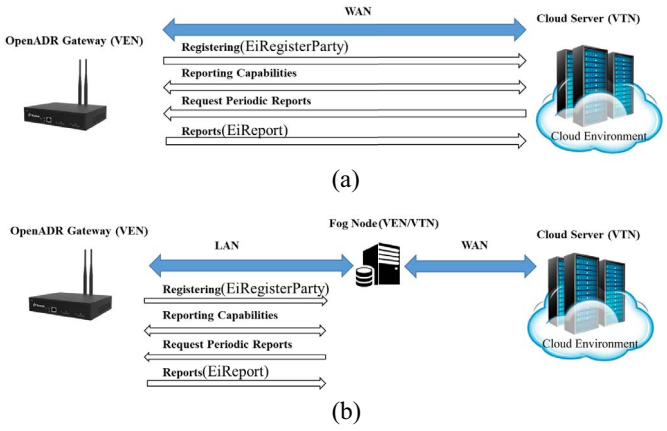


Fig. 2. Message exchange diagram. (a) Cloud-based model. (b) Fog-based model.

all customers registered in the system and send back them to each home gateway.

In the cloud-based model shown in Fig. 2(a), the VTN is located at the cloud server, and all information is transferred to the cloud for further processing and storing. The optimization process is run at the cloud server, and the results are feedbacked to the customers. The cloud computing systems are Internet-based and mainly depend on the Internet connectivity. In this scenario, the home gateways send customer consumption data to the cloud server, resulting in high demand of communication bandwidth especially when the communication channel is not ideal. Suppose $\in$, represents the bit error rate of the communication channel. If the independent channel model employing the Bernoulli function is considered, then the packet error probability $P(S)$ for a packet the size of S is calculated as $P(S) = 1-(1-\in)^S$. Note that as HTTP protocol uses the reliable TCP protocol, the error recovery techniques are utilized for reliable transferring the packet between source and destination. If the packet gets lost in the communication channel, the destination does not receive the acknowledgment signal, and after a period, the sender retransmits the packet. It can be easily proved that for a packet with the length of S, the average number of retransmissions (excluding the original packet) is equal to $(P(S)/[1-P(S)])$. Let $\mathrm{BW}^{\mathrm{Cloud}}$, denotes the total bandwidth requirement for transferring all customer information from all home gateways to the cloud server. $\mathrm{BW}^{\mathrm{Cloud}}$ is calculated as follows:

$$\mathrm{BW}^{\mathrm{Cloud}} = \sum_{i=1}^{|\mathbb{N}|} \frac{S_i}{(1-P(S_i))\triangle t} \frac{b}{s}. \tag{16}$$

As depicted in Fig. 2(b), in the proposed fog-based architecture, the VTN located at the fog nodes in the network edge close to the customers and connected to the customers with high-speed LAN connections. The fog nodes are also capable of aggregating data coming from the home gateways instead of routing all data separately. All power consumption information of customer appliances at each time interval $\triangle t$ is stored in the fog nodes, and just total energy consumption of each customer is forwarded to the cloud server. The optimization process is run at the fog nodes, so it reduces the network traffic. In this scenario, all information needed to run optimization process is exchanged between customers and fog nodes using LAN connection. After that, optimized schedule of each customer's appliances is stored in the local cache of the fog nodes, and only the total optimized power consumption schedule of all customers is aggregated and transferred to the cloud server for further processing. The message size to transfer required information from the fog node to the cloud server, S^{Fog}, is obtained as follows:

$$S^{\mathrm{Fog}} = H_{\mathrm{Size}} + |\mathbb{N}|\ L \tag{17}$$

where L represents the space required to hold each customers' information and his/her power consumption ($|l_i^t|$). The total average required Internet bandwidth in this scenario, $\mathrm{BW}^{\mathrm{Fog}}$, is obtained as follows:

$$\mathrm{BW}^{\mathrm{Fog}} = \frac{S^{\mathrm{Fog}}}{(1-P(S^{\mathrm{Fog}}))\triangle t}\ b/s. \tag{18}$$

G. Delay Analysis

The main goal of this section is to calculate the time elapsed from the moment of transmitting the customer's power consumption data to running the optimization program and sending back of results to the customers, in both cloud- and fog-based models shown in Fig. 2. The total delay of both models consists of two major components including communication delay and computation delay. Suppose $T_{\mathrm{opt}}^{\mathrm{fog}}$, $T_{\mathrm{com}}^{\mathrm{fog}}$, $T_{\mathrm{opt}}^{\mathrm{cloud}}$, and $T_{\mathrm{com}}^{\mathrm{cloud}}$ denote the optimization and communication delay of the fog and cloud models, respectively. Suppose v^{fog} and v^{cloud} represent the fog and cloud server processing speed. As the cloud servers are more powerful and richer than fog nodes, v^{cloud} is much bigger than v^{fog}($v^{\mathrm{fog}} \ll v^{\mathrm{cloud}}$). Suppose $\mathrm{Size}^{\mathrm{opt}}$ represents the size of optimization problem which is equal for both fog- and cloud-based models. The optimization delay of fog and cloud servers are computed as follows:

$$T_{\mathrm{opt}}^{\mathrm{fog}} = \frac{\mathrm{Size}^{\mathrm{opt}}}{v^{\mathrm{fog}}} \quad T_{\mathrm{opt}}^{\mathrm{cloud}} = \frac{\mathrm{Size}^{\mathrm{opt}}}{v^{\mathrm{cloud}}}. \tag{19}$$

Note that as both fog and cloud server run the same optimization problem, we always have $T_{\mathrm{opt}}^{\mathrm{cloud}} \ll T_{\mathrm{opt}}^{\mathrm{fog}}$. On the other hand, as the proposed optimization problem is central, all customer related information should be transferred to the server. According to the optimization process given in (14), each home gateway, based on historical power consumption and generation, should predict the values of $\vec{l}_i^t$, $l_{a,i}^t$, and b_i^t. Furthermore, for each shiftable appliances $a \in \mathbb{A}_i^s$, the values of $S\_\mathrm{time}_i^a$, $E\_\mathrm{time}_i^a$, $E_{a,i}$, $l_{a,i}^{\min}$, and $l_{a,i}^{\max}$ should be determined by the customer. Suppose S_i^{opt} represents the required data size for transferring optimization information of customer i to the server. S_i^{opt} is calculated by the following equation:

$$\begin{aligned} S_i^{\mathrm{opt}} = {} & H_{\mathrm{Size}} + |\mathbb{A}_i^s| \\ & \times \left(|S\_\mathrm{time}_i^a| + |E\_\mathrm{time}_i^a| + |E_{a,i}| + \left|l_{a,i}^{\min}\right| + |l_{a,i}^{\max}|\right) \\ & + |T|\left(\left|\vec{l}_i^t\right| + |\mathbb{A}_i^s||l_{a,i}^t| + |b_i^t|\right). \end{aligned} \tag{20}$$

Note that for both fog- and cloud-based models, the same volume of customer data should be transferred to the corresponding servers but at different communication speed. Suppose for each customer $i \in \mathbb{N}$, $\mathrm{RTT}_i^{\mathrm{fog}}$ and $\mathrm{RTT}_i^{\mathrm{cloud}}$ represent the round-trip time (RTT) between sending the packet and getting its acknowledge, respectively. Let R_{LAN} and R_{WAN} represent the channel speed of LAN and WAN network connections (note that $R_{\mathrm{WAN}} \ll R_{\mathrm{LAN}}$). $\mathrm{RTT}_i^{\mathrm{fog}}$ and $\mathrm{RTT}_i^{\mathrm{cloud}}$ are computed as follows:

$$\mathrm{RTT}_i^{\mathrm{fog}} \cong \frac{2S_i^{\mathrm{opt}}}{R_{\mathrm{LAN}}} \quad \mathrm{RTT}_i^{\mathrm{cloud}} \cong \frac{2S_i^{\mathrm{opt}}}{R_{\mathrm{WAN}}}. \tag{21}$$

As the communication channel is not ideal, the packet may be lost. In this case, after the retransmission timeout (RTO) expires, the packet is retransmitted. Without loss of generality, we assume the RTO is equal to RTT. As the channel is not ideal, for each customer $i = 1, \ldots, N$, the average number of packet transmission (including the original one) is equal to $(1/[(1 - P(S_i^{\mathrm{opt}}))])$. Each packet needs RTT seconds to reach to the destination. So, the average time required to deliver packet to the destination is equal to $(\mathrm{RTT}/[(1 - P(S_i^{\mathrm{opt}}))])$. It should be noted that this delay is not the same for all customers and depends on customer distance, message size and channel bit error rate. As the optimization problem is central, before running the optimization program, all messages containing customer's power consumption information should arrive at their destination. Therefore, in order to calculate $T_{\mathrm{com}}^{\mathrm{fog}}$ and $T_{\mathrm{com}}^{\mathrm{cloud}}$, we must consider the worst case and calculate the maximum communication delay between all customers in the system. So, to compute $T_{\mathrm{com}}^{\mathrm{fog}}$ and $T_{\mathrm{com}}^{\mathrm{cloud}}$, from the N different available values we find the biggest one as follows:

$$T_{\mathrm{com}}^{\mathrm{fog}} = \max\left\{\frac{\mathrm{RTT}_1^{\mathrm{fog}}}{\left(1 - P\left(S_1^{\mathrm{opt}}\right)\right)}, \ldots, \frac{\mathrm{RTT}_N^{\mathrm{fog}}}{\left(1 - P\left(S_N^{\mathrm{opt}}\right)\right)}\right\}$$
$$T_{\mathrm{com}}^{\mathrm{cloud}} = \max\left\{\frac{\mathrm{RTT}_1^{\mathrm{cloud}}}{\left(1 - P\left(S_1^{\mathrm{opt}}\right)\right)}, \ldots, \frac{\mathrm{RTT}_N^{\mathrm{cloud}}}{\left(1 - P\left(S_N^{\mathrm{opt}}\right)\right)}\right\}. \tag{22}$$

By substituting (21) into (22), the total delay of fog- and cloud-based models is calculated as follows:

$$T^{\mathrm{fog}} = \frac{\mathrm{Size}^{\mathrm{opt}}}{v^{\mathrm{fog}}} + \max \frac{2S_i^{\mathrm{opt}}}{\left(1 - P\left(S_i^{\mathrm{opt}}\right)\right)R_{\mathrm{LAN}}}, i = 1, \ldots, N$$
$$T^{\mathrm{cloud}} = \frac{\mathrm{Size}^{\mathrm{opt}}}{v^{\mathrm{cloud}}} + \max \frac{2S_i^{\mathrm{opt}}}{\left(1 - P\left(S_i^{\mathrm{opt}}\right)\right)R_{\mathrm{WAN}}}, i = 1, \ldots, N. \tag{23}$$

Note that the output of optimization program (14) is the optimal power consumption of each customers and is independent of the above communication delay.

IV. Implementation and Simulation Results

In this section, using the simulation and implementation results, we evaluate the performance of the proposed architecture in terms of different power grid and communication network metrics. To collect the customer's home information, an IoT board containing BME280 temperature/humidity/pressure sensor, MQ-2 gas sensor, light sensor, ACS 712 current sensor, 5-V relay, and an ESP 8266 Huzzah MCU/WiFi controller has been implemented. Note that in this paper, we only use the current sensor data. In addition to the HTTP, for efficient data gathering and transmission, the CoAP [42] protocol over WiFi network which is a lightweight protocol and follows the REST has been implemented. We have developed the home gateway on a Raspberry Pi3 with the Raspbian operating system. An HTTP/CoAP forward proxy has been implemented on the home gateway using the python 3.6.2 and CoAPthon [43] python library which supports both CoAP (to communicate with the home appliances) and HTTP (to communicate with the Internet clients). We have implemented a HTTP gateway by JavaFX programming language on Rasbery Pi3. We have used the ThingsSpeak open IoT platform [44] for our data collecting and storage and the OpenADR 2.0b VEN and VTN [45]. The implementation testbed and the communication protocol stack are shown in Fig. 3. As it can be seen, the home gateway communicates with the IoT node through HTTP/CoAP protocol. It provides a Web interface for HTTP client to access the IoT resources through a user interface. As CoAP message is different with the HTTP, an HTTP to CoAP mapping is designed to provide the necessary functionalities.

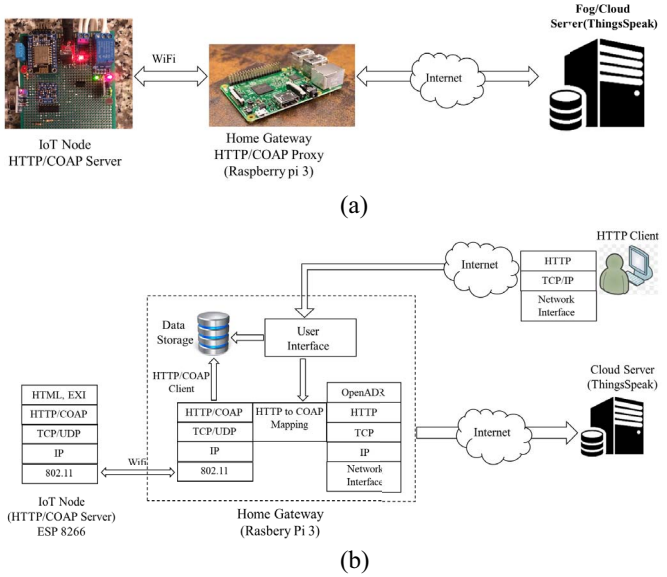


Fig. 3. (a) Implementation test bed. (b) Protocol architecture and layering.

Fig. 4 shows the detail algorithm of the home gateway in the HTTP operation mode. As it is shown in this figure, at a regular time interval (15 s), the gateway makes an HTTP connection to the IoT node and send the GET method. The GET method contains the URI of the requested resources. After that, the gateway gets the response from the sensor node. If the response status is 200OK, it means that the connection has been established successfully. Then the payload of the response message which contains the sensor data is extracted from the message. Now, the gateway makes an HTTP connection to the fog/cloud node and sends the POST method which

While True:
1. *do*
   a. Make an HTTP connection to the IoT node.
   b. Send "GET" request method
   c. R1= get response from the IoT node
2. *while* (R1.status !=200 OK)
3. Read senores data
4. data=R1.read()
5. *do*
   a. Make an HTTP connection to the fog node/cloud server
   b. Send "GET" request method
   c. R2= get response from the server
6. *while* (R2.status !=200 OK)
7. Put sensor data in the POST payload
8. Send "POST" method to the fog/cloud server

Fig. 4. Home gateway algorithm (HTTP protocol).

contains the sensor data to the server. Then both connections are closed. This procedure is repeated each 15 s, periodically.

The same procedure is applied for CoAP protocol. The only difference is that as CoAP uses the user datagram protocol (UDP) protocol as transport layer protocol which is unreliable, there is no guaranty for a successful connection. For this reason, the gateway is programmed to send four consequence requests in the hope that one of them will reach the sensor node successfully.

A. Home Monitoring Results

Fig. 5(a) shows the temperature, humidity and the power consumption of a refrigerator and a LCD monitor for a home in Toronto, CA, USA, during 12 A.M. to 16 P.M. on August 17, 2017, which were gathered by the IoT node. These data are captured by the IoT node and sent to the home gateway. However, in the current implementation just the power consumption information is processed and forwarded to the fog node. To evaluate and compare the performance of CoAP and HTTP protocols, we implemented an HTTP and CoAP servers on two different IoT nodes. The home gateway starts to communicate with these nodes, every 15 s and captures 50 consequence samples of sensed data.

In Fig. 6, the bandwidth consumptions of both protocols are plotted versus time. It can be observed that HTTP consumes more bandwidth than CoAP. This is because the HTTP uses the TCP as its underlying transport protocol, which needs more packet communication than CoAP which is a lightweight protocol based on connection less UDP protocol. Table I presents the performance comparison of HTTP and CoAP in terms of the number of GET, ACK, and IP packets, average bandwidth consumption and the packet length distribution. It can be seen that both protocols send 50 different GET messages and receive 50 ACK messages containing the gathered sensed data. As for each HTTP GET message a TCP connection should be established and different TCP segments should be interchanged between the end nodes, so HTTP protocol consumes more bandwidth and transfers more IP packets with different sizes.

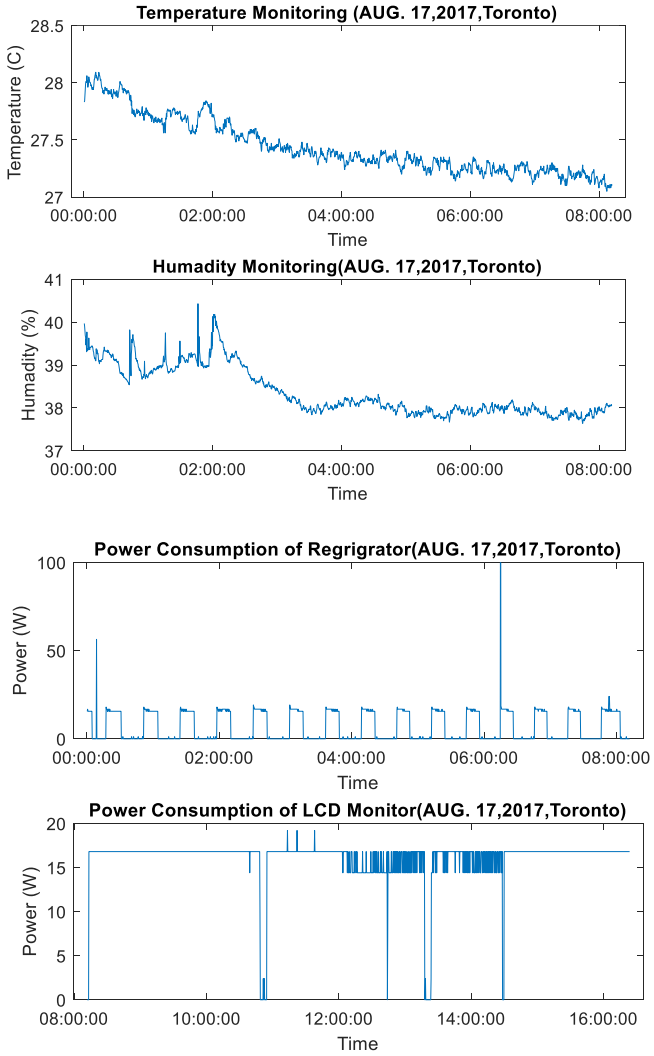


Fig. 5. Temperature, humidity, and power consumption monitoring.

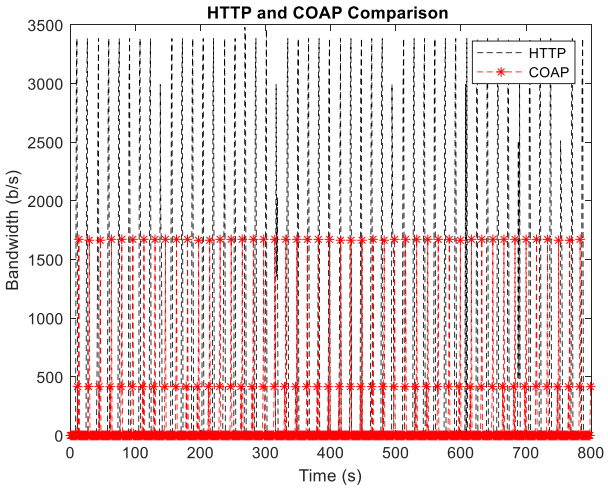


Fig. 6. HTTP and CoAP bandwidth consumptions for 50 samples.

B. Power Consumption Prediction

We implemented the NLMS predictor to predict power consumption of some customers using the UCI Machine Learning

TABLE I
HTTP AND COAP PROTOCOL COMPARISON

		HTTP	COAP
Number of GET messages		50	50
Number of ACK messages		50	50
#of IP packet		502	102
Average bandwidth (b/s)		390	123
Packet length distribution Percentages (Bytes)	0-39	0	0
	40-79	%69.7	%50
	80-159	%20.3	0
	160-319	%10	%50
	320-higer	0	0

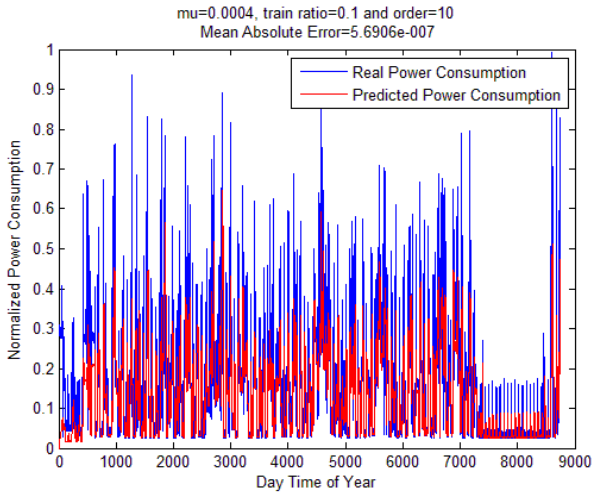


Fig. 7. Power consumption prediction results.

Repository individual household electric power consumption data set [46]. Fig. 7, for historical records size equal to 8760, with $\mu = 0.0004$ and $p = 10$ and train ratio $= 10\%$, depicts the predicted power consumption. Results confirm that the value of mean absolute error is 5.69e-7.

C. Power Grid Performance

In this section, we evaluate the performance of the proposed TE-based DR program. For this purpose, we have simulated a community with 100 different customers. Each customer's home has equipped by a PV system with random solar panel size between 10–30 m^2, a battery, a PEV and a random number of appliances.

We use time of use pricing where during 11 A.M.–17 P.M. there is peak load condition and grid energy price is equal to 20 cents/KWh and the other period, it is nonpeak hours and grid energy price is equal to 10 cents/KWh. We suppose that 20% of customers are *energy sellers* and the others are *energy buyers*. The *energy sellers* have been equipped with a solar system with a total capacity of twice the *energy buyers*. We set a_t is equal to 0.1 and 0.05 cents during peak and nonpeak hours, respectively. The R_{WAN} and R_{LAN} are set to 10 Mb/s and 1 Gb/s, respectively.

Fig. 8 shows the total shiftable (including PEV) and non-shiftable loads in the power system. The shiftable load has been set to almost 40% nonshiftable load.

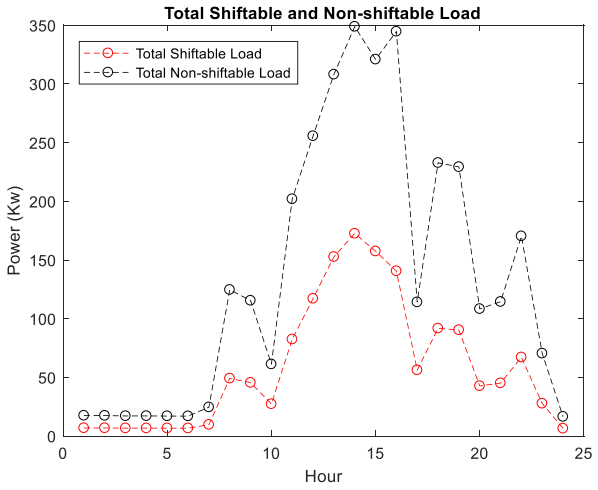


Fig. 8. Total shiftable (including PEV) and nonshiftable loads in the power system.

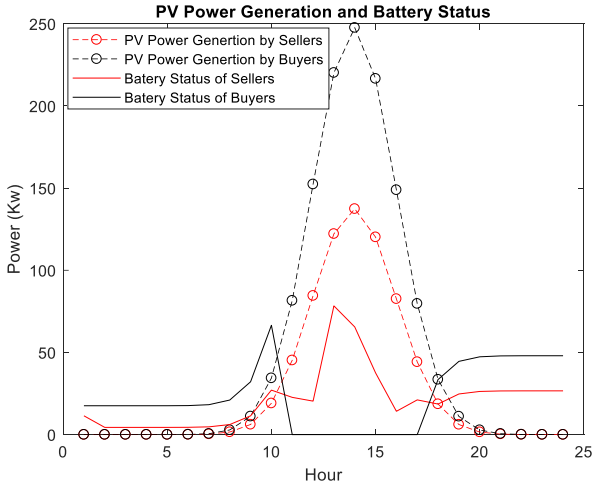


Fig. 9. PV generation and battery status of *energy buyers* and *energy sellers*.

Fig. 9, for both *energy sellers* and *energy buyers*, shows the amount of PV energy generation and the battery status during a particular day. It can be seen that during peak hours, *energy buyers* consume all energy stored in their battery, so their battery SoC is always zero. Unlike the *energy buyers*, the *energy sellers* always have stored energy in their battery to sell it to the *energy buyers*.

In Fig. 10(a), the total energy requested by *energy buyers* (E_B^t) and the total energy for sale by *energy sellers* (E_S^t) are depicted versus time. Fig. 10(b) shows the price rate of both power grid (pr_G) and proposed TE system (pr_{TE}). It can be seen that during peak hour time, unlike pr_G, pr_{TE} changes with the change of total demand and amount of supply. By looking at these graphs, we conclude that during time 13 P.M.–15 P.M., where the difference between demand and supply ($E_B^t - E_S^t$) is small, pr_{TE} goes down, while at time 16 P.M. where the demand is much more than supply, pr_{TE} goes up.

In Fig. 11, the total daily optimized and unoptimized load (with and without PV) are plotted versus the time of day. As it can be seen in the figure, the optimization process shifts

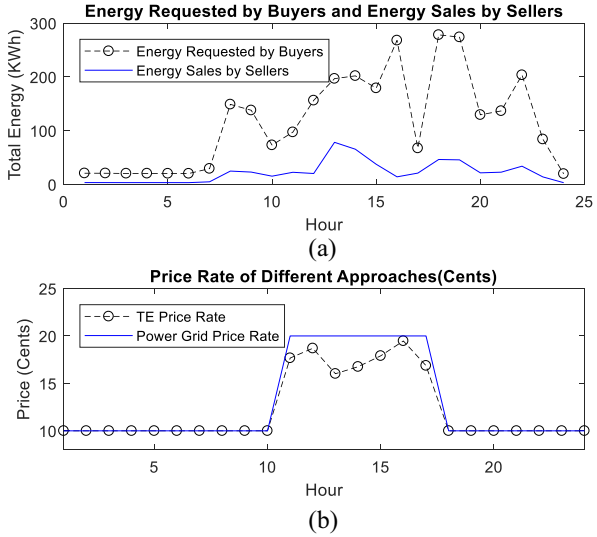


Fig. 10. (a) Energy demand of *energy buyers*. (b) TE and grid price.

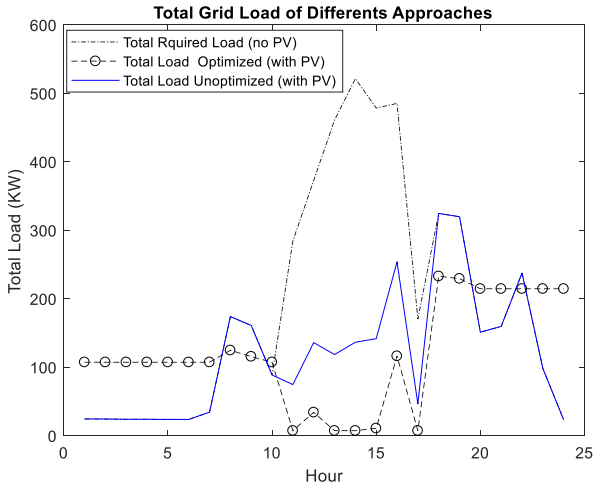


Fig. 11. Total demand before and after optimization.

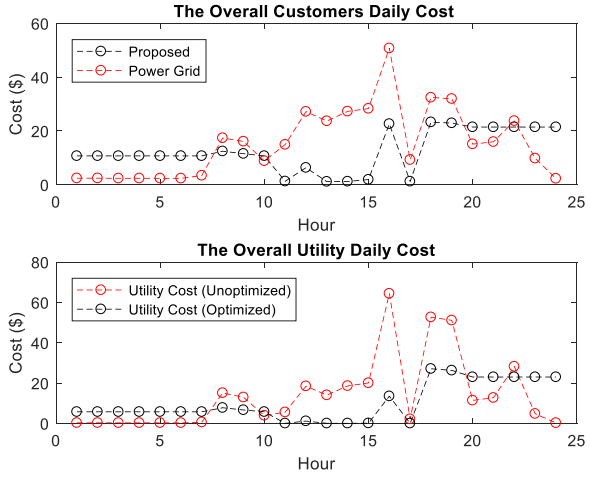


Fig. 12. Overall cost of customers and utility company before and after optimization.

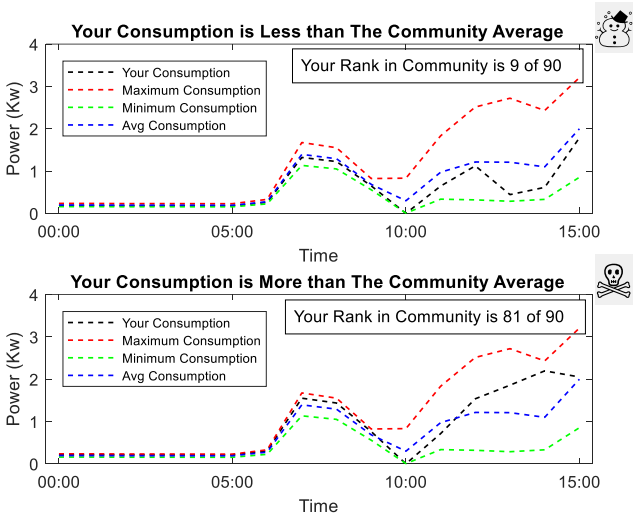


Fig. 13. Consumption statics of the community and two different customers.

some unnecessary consumption from peak hours to the non-peak hours to decrease customer costs. Results confirm that, by using the proposed optimization program, the peak to average ratio which is defined as the maximum daily load divided by the average load, is reduced by 0.79.

Fig. 12 depicts the overall daily cost of both customers and utility company using the proposed and power grid pricing. It is clear that not only customers but also the utility company benefit from the proposed optimization program. Customers can reduce their daily cost by as much as 74$ and the utility company also save 96$ a day. As we assumed there are only 100 customers in the power grid, the average daily benefit for customers and the utility company is almost 74 and 96 cents per customer, respectively.

In Fig. 13, for two different customers with different consumptions, the user interface which plot the average, maximum, minimum, and user consumption is shown at time $t = 15$ P.M. This information is prepared by the fog nodes and forwarded to each customer's home gateway. Each customer uses this information to evaluate his/her consumption regarding the maximum, minimum, and average consumption in the community. As the fog node has the power consumptions statics of all customers in the community, it can compute and inform the rank of each customer in terms of total power consumptions. High power consumer customers may improve their energy consumption based on their positions in the community.

D. Communication Performance

In this section, we evaluate the communication performance of the cloud- and fog-based models shown in Fig. 2. Fig. 14, for both models, and at two different values of channel bit error rate, plots the average message size, the number of retransmissions and the bandwidth versus simulation time. It can be seen that for both cases, the fog model has better bandwidth performance than the cloud model. As shown in this figure, the message size of fog-based model is always more than cloud-based model (900 bytes versus almost 575 bytes). This is because, in the fog-based model, all customers information is aggregated at the fog node and forwarded to the cloud server

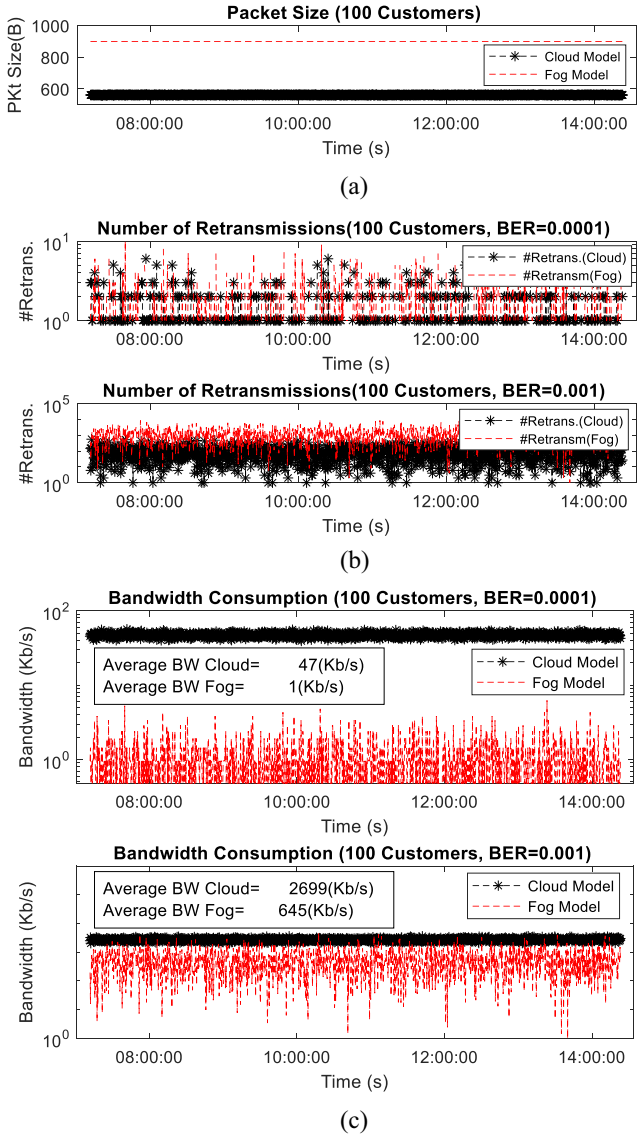


Fig. 14. (a) Packet size. (b) Number of retransmissions. (c) Total bandwidth consumptions of the fog- and cloud-based models at two different values of channel bit error rate.

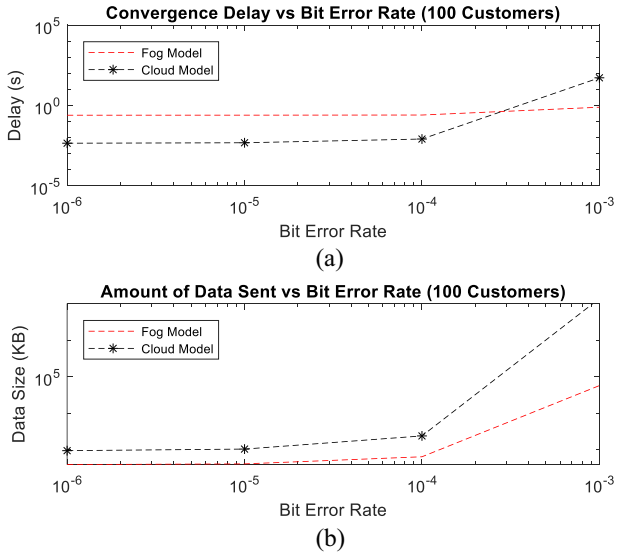


Fig. 15. (a) Convergence delay. (b) Required bandwidth versus channel bit error rate.

through a single HTTP message. So, the fog-based model has a higher number of retransmissions with more fluctuations than the cloud-based model. But as the fog node aggregates all customers' information in a single HTTP message, its bandwidth performance is much better than the cloud-based model.

Results confirm that by increasing the channel bit error rate, the number of retransmission and the total bandwidth is also increased. In the next simulation trials, the convergence delay and the total data size for running optimization problem is plotted versus communication bit error rate. We assume the cloud server is 100 times faster than the fog nodes ($\nu^{\text{cloud}} = 100\nu^{\text{fog}}$). Results are displayed in Fig. 15. Fig. 15(a) confirms that at low bit error rate, where the number of retransmissions is low, the communication delay is low as well. In this case, as the cloud server has more computation capabilities than the fog nodes, its computation delay is less than the fog node. So, the delay convergence of cloud-based model is less than fog-based model. On the other hands, by increasing the channel bit error rate, the communication delay of cloud-based model also increases. Results shown in Fig. 15(b) confirm that when bit error rate increases, both cloud- and fog-based models require more bandwidth. However, as the fog-based model aggregates all customer information, its bandwidth consumption is always less than the cloud-based model. Note that we set the maximum message size to 1500 bytes. In case, when total message size is more than 1500 bytes, the message is fragmented to the smaller messages.

V. Conclusion

In this paper, we presented a multitier communication architecture for TE management systems. The architecture includes home gateways, local fog nodes and cloud server. The home gateway communicates with the IoT node to collect sensor data. We implemented CoAP/HTTP IoT nodes and HTTP/CoAP forward proxy on a home gateway which collects consumption information and the other related data from the customer home and transferred it to the servers through a home gateway. The OpenADR protocol has been implemented on the home gateway which provides customers useful information such as the instantaneous amount of energy uses, the power consumption status relative to the average, maximum, and minimum consumption of the community, the power consumption of each electric device and the optimized day ahead schedule of customer's appliances. This information helps customers to take intelligent decision for their energy consumption. The fog nodes act as retail energy market server which provides energy services to the end-users. We proposed an intercustomer energy trading cost function. In the proposed system, customers prefer to buy energy from each other rather than buying from the power grid which is always more expensive at the peak load times. We also proposed a TE-based DR program which employed a day ahead optimization to schedule customer appliances. The optimization program has been defined to benefit not only customers but also utility company. We evaluated the bandwidth requirements and

delay performance of the proposed fog-based model and compared it with the cloud-based model. Results confirmed that the fog-based model significantly decreases the total bandwidth and delay especially when the communication channel is not ideal.

REFERENCES

[1] *GridWise Transactive Energy Framework Version 1.0*. Grid-Wise Archit. Council, U.S. Dept. Energy, Washington, DC, USA, 2015.

[2] D. J. Hammerstrom, "Pacific northwest smart grid demonstration transactive coordination signals," Battelle—Pac. Northwest Div., Richland, WA, USA, Rep. PNWD-4402 Rev X, 2013.

[3] D. J. Hammerstrom *et al.*, "Pacific northwest smart grid demonstration project technology performance report, volume 1: Technology performance," Battelle—Pac. Northwest Div., Richland, WA, USA, Rep. PNWD-4445, 2015.

[4] Pacific Northwest National Laboratory. (2017). *VOLTTRON^{TM}—An Intelligent Agent Platform for the Smart Grid*. [Online]. Available: http://gridoptics.pnnl.gov/VOLTTRON/

[5] J. Hagerman, *EERE & Buildings to Grid Integration*. DOE Build. Technol. Office, Washington, DC, USA, 2015.

[6] S. Widergren, J. Fuller, C. Marinovici, and A. Somani, "Residential transactive control demonstration," in *Proc. IEEE PES Innov. Smart Grid Technol. Conf.*, Washington, DC, USA, Feb. 2014, pp. 1–5.

[7] D. Forfia, M. Knight, and R. Melton, "The view from the top of the mountain: Building a community of practice with the GridWise transactive energy framework," *IEEE Power Energy Mag.*, vol. 14, no. 3, pp. 25–33, May/Jun. 2016.

[8] (2017). *ERCOT Real-Time Market*. [Online]. Available: http://www.ercot.com/mktinfo/rtm

[9] *NYISO Markets & Operations*, New York Independent Syst. Oper., New York, NY, USA, 2010.

[10] *CAISO Market Processes and Products*, California Independent Syst. Oper., Folsom, CA, USA, 2017. [Online]. Available: http://www.caiso.com/Pages/default.aspx

[11] N. Ruiz, I. Cobelo, and J. Oyarzabal, "A direct load control model for virtual power plant management," *IEEE Trans. Power Syst.*, vol. 24, no. 2, pp. 959–966, May 2009.

[12] "Open automated demand response communications specification (version 1.0)," California Energy Commission, Sacramento, CA, USA, Rep. CEC-500-2009-063, Apr. 2009.

[13] E. Baccarelli *et al.*, "Fog of everything: Energy-efficient networked computing architectures, research challenges, and a case study," *IEEE Access*, vol. 5, pp. 9882–9910, 2017.

[14] I. Stojmenovic and S. Wen, "The fog computing paradigm: Scenarios and security issues," in *Proc. Federated Conf. Comput. Sci. Inf. Syst. (FedCSIS)*, Warsaw, Poland, 2014, pp. 1–8.

[15] S. V. Vandebroek, "1.2 three pillars enabling the Internet of everything: Smart everyday objects, information-centric networks, and automated real-time insights," in *Proc. IEEE Int. Solid State Circuits Conf. (ISSCC)*, San Francisco, CA, USA, 2016, pp. 14–20.

[16] P. G. V. Naranjo *et al.* (2017). *FOCAN: A Fog-Supported Smart City Network Architecture for Management of Applications in the Internet of Everything Environments*. [Online]. Available: https://arxiv.org/pdf/1710.01801.pdf

[17] P. G. V. Naranjo *et al.*, "Big data over smartgrid—A fog computing perspective," in *Proc. SOFTCOM Workshop*, 2016, pp. 1–6.

[18] R. Deng, R. Lu, C. Lai, and T. H. Luan, "Towards power consumption-delay tradeoff by workload allocation in cloud-fog computing," in *Proc. IEEE Int. Conf. Commun. (ICC)*, London, U.K., 2015, pp. 3909–3914.

[19] F. Jalali, A. Vishwanath, J. de Hoog, and F. Suits, "Interconnecting fog computing and microgrids for greening IoT," in *Proc. IEEE Innov. Smart Grid Technol. Asia (ISGT-Asia)*, Melbourne, VIC, Australia, 2016, pp. 693–698.

[20] Y. Yan and W. Su, "A fog computing solution for advanced metering infrastructure," in *Proc. IEEE/PES Transm. Distrib. Conf. Exposit. (T&D)*, Dallas, TX, USA, 2016, pp. 1–4.

[21] T. Yashiro, S. Kobayashi, N. Koshizuka, and K. Sakamura, "An Internet of Things (IoT) architecture for embedded appliances," in *Proc. IEEE Region 10 Humanitarian Technol. Conf. (R10-HTC)*, 2013, pp. 314–319.

[22] D.-M. Han and J.-H. Lim, "Design and implementation of smart home energy management systems based on ZigBee," *IEEE Trans. Consum. Electron.*, vol. 56, no. 3, pp. 1417–1425, Aug. 2010.

[23] N. Langhammer and R. Kays, "Performance evaluation of wireless home automation networks in indoor scenarios," *IEEE Trans. Smart Grid*, vol. 3, no. 4, pp. 2252–2261, Dec. 2012.

[24] V. Tanyingyong, R. Olsson, J.-W. Cho, M. Hidell, and P. Sjodin, "IoT-grid: IoT communication for smart DC grids," in *Proc. IEEE Glob. Commun. Conf. (GLOBECOM)*, Washington, DC, USA, 2016, pp. 1–7.

[25] E. Spanò, L. Niccolini, S. D. Pascoli, and G. Iannacconeluca, "Last-meter smart grid embedded in an Internet-of-Things platform," *IEEE Trans. Smart Grid*, vol. 6, no. 1, pp. 468–476, Jan. 2015.

[26] D. Minoli, K. Sohraby, and B. Occhiogrosso, "IoT considerations, requirements, and architectures for smart buildings—Energy optimization and next-generation building management systems," *IEEE Internet Things J.*, vol. 4, no. 1, pp. 269–283, Feb. 2017.

[27] F. G. Brundu *et al.*, "IoT software infrastructure for energy management and simulation in smart cities," *IEEE Trans. Ind. Informat.*, vol. 13, no. 2, pp. 832–840, Apr. 2017.

[28] U. Herberg, D. Mashima, J. G. Jetcheva, and S. Mirzazad-Barijough, "OpenADR 2.0 deployment architectures: Options and implications," in *Proc. IEEE Int. Conf. Smart Grid Commun. SmartGridComm*, Venice, Italy, 2014, pp. 782–787.

[29] *IEEE Standard for Information Technology, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications—Amendment 4: Enhancements for Very High Throughput for Operation in Bands Below 6 GHz*, IEEE Standard 802.11ac(TM)-2013, pp. 1–425, Dec. 18, 2013.

[30] *Approved Amendment to Standard for Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks Specific Requirements—Part 15.3: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for High Rate Wireless Personal Area Networks (WPAN) Amendment to Mac Sublayer*, IEEE Standard P802.15.3b/D04, 2005.

[31] *IEEE Standard for Low-Rate Wireless Networks*, IEEE Standard 802.15.4-2015, pp. 1–709, Apr. 22, 2016.

[32] OPOWER. (2016). *Demand Response*. [Online]. Available: https://opower.com/products/demand-response/

[33] R. G. Garroppo, S. Giordano, M. Pagano, and G. Procissi, "On traffic prediction for resource allocation: A Chebyshev bound based allocation scheme," *Comput. Commun.*, vol. 31, no. 16, pp. 3741–3751, 2008.

[34] I. Atzeni, L. G. Ordonez, G. Scutari, D. P. Palomar, and J. R. Fonollosa, "Noncooperative and cooperative optimization of distributed energy generation and storage in the demand-side of the smart grid," *IEEE Trans. Signal Process.*, vol. 61, no. 10, pp. 2454–2472, May 2013.

[35] D. T. Nguyen and L. B. Le, "Optimal energy management for cooperative microgrids with renewable energy resources," in *Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm)*, Vancouver, BC, Canada, Oct. 2013, pp. 678–683.

[36] M. P. Fanti, A. M. Mangini, M. Roccotelli, and W. Ukovich, "Optimal energy management integrating renewable energy, energy storage systems and electric vehicles," in *Proc. IEEE 14th Int. Conf. Netw. Sens. Control (ICNSC)*, Calabria, Italy, 2017, pp. 519–524.

[37] M. E. Khodayar, L. Wu, and M. Shahidehpour, "Hourly coordination of electric vehicle operation and volatile wind power generation in SCUC," *IEEE Trans. Smart Grid*, vol. 3, no. 3, pp. 1271–1279, Sep. 2012.

[38] A.-H. Mohsenian-Rad, V. W. S. Wong, J. Jatskevich, R. Schober, and A. Leon-Garcia, "Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid," *IEEE Trans. Smart Grid*, vol. 1, no. 3, pp. 320–331, Dec. 2010.

[39] A. J. Wood and B. F. Wollenberg, *Power Generation, Operation, and Control*. New York, NY, USA: Wiley, 1996.

[40] A. Nemirovski, "Interior point polynomial time methods in convex programming," Lecture Notes, Georgia Inst. Technol., School Ind. Syst. Eng., Atlanta, GA, USA, ISYE 8813, 2004.

[41] S. P. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press. 2004, p. 143.

[42] Z. Shelby, K. Hartke, and C. Bormann, "The constrained application protocol (CoAP)," Internet Eng. Task Force, Fremont, CA, USA, RFC 7252, Jun. 2014.

[43] (2017). *CoAPthon Python Library*. [Online]. Available: https://github.com/Tanganelli/CoAPthon

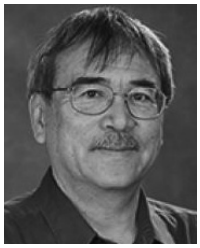
[44] (2017). *ThingSpeak Open IOT Plateform*. [Online]. Available: https://thingspeak.com/

[45] (2017). *Certified EPRI Open Source OpenADR 2.0b VEN & VTN*. [Online]. Available: http://www.openadr.org/epri-certified-open-source-code

[46] (2012). *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Individual+household+electric+power+consumption

Mohammad Hossein Yaghmaee Moghaddam (SM'12) received the B.S. degree from the Sharif University of Technology, Tehran, Iran, in 1993, and the M.S. and Ph.D. degrees in communication engineering from the Tehran Polytechnic (Amirkabir) University of Technology, Tehran, in 1995 and 2000, respectively.

From 1998 to 1999, he was a Visiting Research Scholar with the Network Technology Group, C&C Media Research Laboratories, NEC Corporation, Tokyo, Japan. From 2007 to 2008, he was a Visiting Associate Professor with the Lane Department of Computer Science and Electrical Engineering, West Virginia University, Morgantown, WV, USA. From 2015 to 2016, he was a Visiting Professor with the Electrical and Computer Engineering Department, University of Toronto, Toronto, ON, Canada. He is currently a Full Professor with the Computer Engineering Department, Ferdowsi University of Mashhad, Mashhad, Iran. His current research interests include smart grid communication, software defined networking, and network function virtualization.



Alberto Leon-Garcia (F'99) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of Southern California, Los Angeles, CA, USA, in 1973, 1974, and 1976, respectively.

He was the Founder and the CTO of AcceLight Networks, Ottawa, ON, Canada, from 1999 to 2002, which developed an all-optical fabric multiterabit, switch. He is currently a Professor of electrical and computer engineering with the University of Toronto, Toronto, ON, Canada. He authored the textbook *Probability and Random Processes for Electrical Engineering and Communication Networks: Fundamental Concepts and Key Architecture*. His current research interests include application-oriented networking and autonomic resources management with a focus on enabling pervasive smart infrastructure.

Prof. Leon-Garcia was a recipient of the 2006 Thomas Eadie Medal from the Royal Society of Canada and the 2010 IEEE Canada A. G. L. McNaughton Gold Medal for his contributions to the area of communications. He is a Fellow of the Engineering Institute of Canada.