Intelligent Edge Computing for IoT-Based Energy Management in Smart Cities

Yi Liu, Chao Yang, Li Jiang, Shengli Xie, and Yan Zhang

Abstract

In recent years, green energy management systems (smart grid, smart buildings, and so on) have received huge research and industrial attention with the explosive development of smart cities. By introducing Internet of Things (IoT) technology, smart cities are able to achieve exquisite energy management by ubiquitous monitoring and reliable communications. However, long-term energy efficiency has become an important issue when using an IoT-based network structure. In this article, we focus on designing an IoT-based energy management system based on edge computing infrastructure with deep reinforcement learning. First, an overview of IoT-based energy management in smart cities is described. Then the framework and software model of an IoT-based system with edge computing are proposed. After that, we present an efficient energy scheduling scheme with deep reinforcement learning for the proposed framework. Finally, we illustrate the effectiveness of the proposed scheme.

Introduction

The global amount of urban population is larger than 50 percent, and is expected to rise to 70 percent by 2050 [1]. To deal with the explosive increment of population, smart city projects are proposed by many countries and organizations for promoting and developing a new paradigm to optimize energy consumption in cities. Due to the complexity of the energy management environment, which includes distribution networks, households/buildings (power and heating), and so on [2, 3], numerous types of information need to be transferred in real time. For instance, electricity peak smoothing requires information about energy profiles as well as users' acceptable level. These requirements prompt the smart city to employ new information and communication technologies, such as Internet of Things (IoT) communication systems [4, 5], to monitor and transmit data to utility control centers so that complex policies can be implemented for energy management in smart cities.

Since IoT-based systems are widely used in smart cities where large amounts of data are generated and transferred, it is difficult to precisely realize the data from a complex environment and provide reliable control actions accordingly. The emerging deep reinforcement learning (DRL) can be considered as a promising technology, which takes a long-term goal into account and can generate (near-)optimal control actions to time-variant dynamic systems [6]. In the DRL scheme, two phases are included:

- The offline deep neural network (DNN) construction phase, which correlates the value function with corresponding states and actions
- The online dynamic deep Q-learning phase, which contains action selection, system control, and dynamic network updating

To achieve the DRL performance, data transferring between the DRL agent (e.g., cloud server) and devices should be guaranteed. Considering the wireless connections between most devices and the DRL agent, the large amount of data transferring may easily exceed the limit of transmission capacity. Hence, resource-limited IoT devices cannot directly send the DRL requirement to the DRL agent, introducing a major challenge in energy management application.

To deal with this challenge, edge computing (EC) is used as another important technology for IoT-based energy management systems [7]. As EC can provide the computing services at the network "edge" near IoT devices, two major advantages of EC make it suitable for DRL usage in IoT-based systems. First, EC can enormously reduce transmission data from devices to DRL agent by preprocessing procedures. Second, EC is able to bridge the gap between the limited capability of low-powered devices and the computational demand of energy management [8].

In this article, we introduce an IoT-based energy management system that deploys edge computing with DRL. This system can improve the energy management performance as well as reduce the execution time. To easily demonstrate the proposed system, the basic infrastructure and software model are presented. Based on this infrastructure, we specifically state an energy scheduling problem for demand-side response in the smart grid scenario. We design two types of scheduling methods with the DRL algorithm by considering the limitations of EC processing capacity. The experimental results show that our solution outperforms other energy scheduling methods based on IoT systems.

The remainder of this article can be outlined as follows. In the following section, an overview of IoT-based energy management in smart cities is described. The framework and software

Yi Liu, Chao Yang, Li Jiang, and Shengli Xie are with Guangdong University of Technology; Yan Zhang is with University of Oslo.

Digital Object Identifier: 10.1109/MNET.2019.1800254

Based on the information and communication technologies exploited in an IoT system, multi-energy networks can integrate the smart electric grid, supply grids of heat and gas, and network traffic for unified energy management for smart cities.

model of the IoT-based system with edge computing are then proposed. Next, we present an efficient energy scheduling scheme with DRL for the proposed framework. We show the illustrative results of the proposed scheme and then conclude the article.

Overview of IoT-Based Energy Management in Smart Cities

An IoT-based energy management system is a crucial component in building smart city architectures, as it not only enables new energy-related value-added services but also intelligently facilitates the integration of various energy sources and automatic operation control. In this section, we discuss some typical IoT-based energy management systems in a smart city.

Smart Buildings

Smart buildings are expected to be intelligent and humanized, and the data generated by buildings is well processed and utilized [9, 10]. The energy management in smart buildings can be divided into three levels: the device level, system level, and inter-system level.

Device Level: The devices in smart buildings can extend access to the input data (consumed electricity, temperature, humidity, etc.), relevant for the situational awareness about its operational conditions. Specifically, for energy management, the IoT processing unit should process the energy data in a predictive or adaptive manner and transmit the output to the relevant command and control units. Then the final actuators can carry out the needed actions.

System Level: At the system level, the major function of smart building is to coordinate and synchronize the behaviors of various energy-related components. To better manage and control energy-related units, the smart building can automatically reconfigure the internal basic system of the building supported by the IoT system in the short term. The energy-related applications are also supported by many intelligent IoT devices deployed throughout the system.

Inter-System Level: Inter-system level refers to the integration of several autonomous, self-contained subsystems, working together for the sake of fulfilling energy management for smart buildings. In this sense, a smart building is an inter-system in which electricity, gas, heating, and other systems are integrated as a unified system. Its smartness denotes the capacity to manage, interconnect, and adapt various assets and functionalities (including technical, economic, and social factors) for the satisfaction of energy management.

Smart Power Grid

The smart power grid is a key component of the strategies toward a sustainable energy future [11–13]. Driven by IoT usage, they can not only facilitate the integration of renewable energy sources and the electrification of transport, but also provide new energy-related value-added services. In particular, the transformation of energy management will be driven by the design and deployment of smart grids. Smart power grids have the potential to expand their capabilities with smartness and data flows. With IoT assistance, the energy management in smart power grids is able to reach every corner of a city and will stimulate intelligent grid infrastructure and friendly user interaction.

Smart energy plus Internet is a new development of power management systems with deep fusion of the Internet and energy production, transmission, storage, consumption, and market. Its main characteristics include equipment being smart, multiple energy types being synergistic, information being symmetric, supply and demand being distributed, the system being flat, and transactions being open [12]. For example, develop smart energy production and consumption facilities including smart coal mines, smart wind farms, smart PV stations, and so on, as well as Internet-based smart operation cloud platforms, so as to realize remote control optimization and improve operation efficiency and benefit.

Multi-Energy Networks

IoT-based multi-energy networks can improve the overall efficiency and benefit of the energy system in different size areas including large buildings, parks, islands, towns, and so on [14]. Based on the information and communication technologies exploited in an IoT system, multi-energy networks can integrate the smart electric grid, supply grids of heat and gas, and network traffic for unified energy management for smart cities. Furthermore, the benefits of using IoT in a multi-energy system are listed below.

Promoting Deep Fusion of Energy and Information Infrastructure: Based on IoT technology, a multi-energy system can coordinate the construction of energy grids including electric, gas, cool air, heat, and so on, with their information infrastructure including information architecture, storage unit, and so on. Thereby, the information system and energy system, in measurement, computation, control, and so on, can be very efficiently integrated. The integration can motivate the standardize multi-energy network structure and information interface.

Developing Smart Energy Management: Through the development of advanced IoT networks, terminal facilities, including smart homes, smart districts, and so on, can build energy-efficient monitoring platforms, provide personalized energy management/saving service, and realize smart customization and flexible transaction of energy.

Cultivating the Emerging Energy Market and Business Models: By using an IoT system, the multi-energy entities can achieve remote automatic collection and reading of water, gas, heat, and electricity in all-in-one meters. This advantage of an IoT-based energy system can incubate an open and shared market system that includes small and micro users such as individuals, families, and distributed energy resources to share and trade energy flexibly and equally through unified platforms.

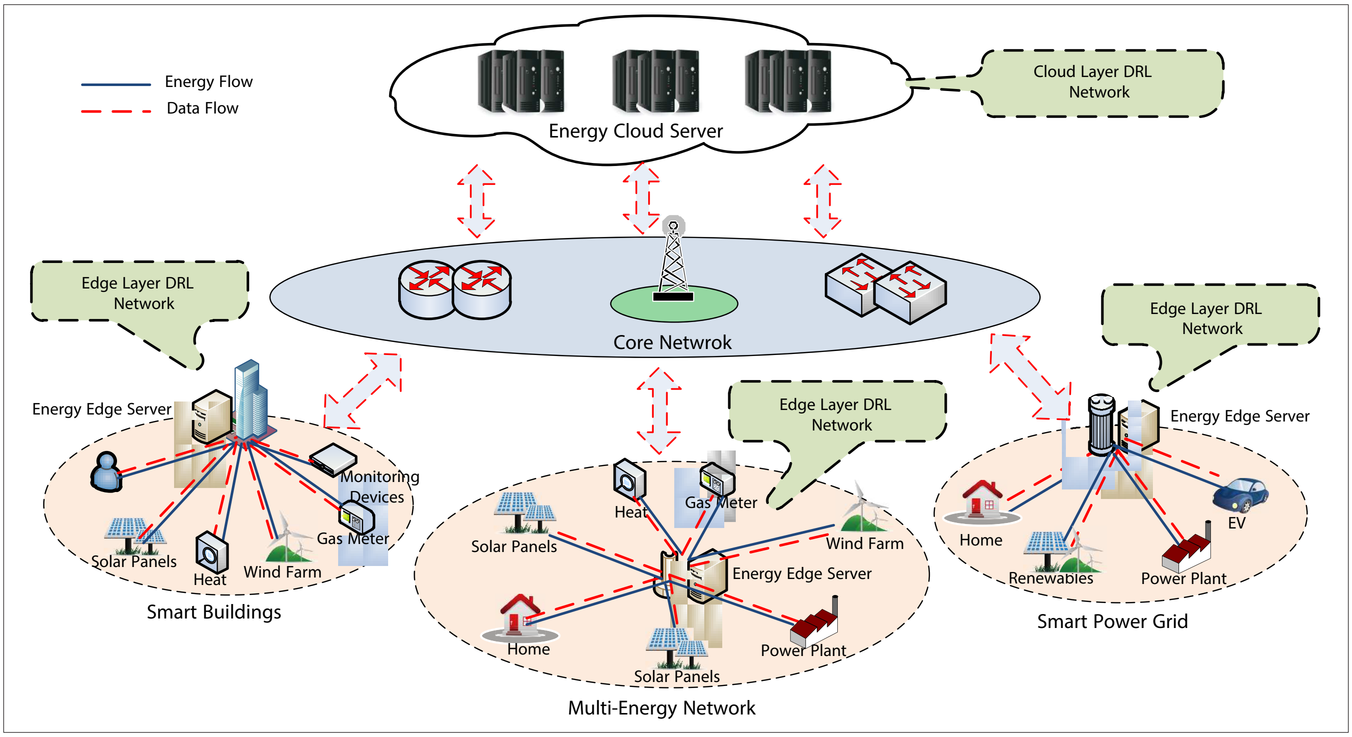

FIGURE 1. The IoT-based energy management architecture for smart cities.

Framework and Software Model for an IoT-Based Energy Management System

IoT-Based Architecture with Intelligent Edge Computing

Figure 1 shows the IoT-based energy management architecture with edge computing based on a DRL network. The architecture consists of three main components: *energy devices*, *energy edge servers*, and *energy cloud servers*.

Energy Device: The energy device can be any entity, device, or user that can supply and require energy in the network. The devices can detect/collect/generate the energy data according to their types.

Energy Edge Server: The energy edge server can be deployed at the network gateway, base station, and so on for computing/caching/delivering energy data in a local area network. It connects to the energy devices by different communication technologies, for example, 5G, WiFi, and vehicular ad hoc network (VANET). The energy edge server can also decide the operations of a local energy network according to the analytical results.

Energy Cloud Server: The energy cloud server connects to the central controller for energy management. The responsibility of the energy cloud server is to not only provide real-time analysis and calculation to energy devices, but also satisfy the computation requirements from energy edge servers.

In the proposed architecture, energy edge servers process the collected data and transmit the data to the cloud server through the core network. In both the cloud server and edge server, the DRL agents are deployed. Upon the energy device having a computation task, it will send the task to a nearby edge server in which the edge DRL agent is responsible for computing the task. For saving energy of the edge server, we can pretrain DNNs in the energy cloud server. After the training phase, we send the DNN weights to the energy edge server, which operates the deep Q-learning process. In this case, the energy edge servers load the data from the energy devices and then transfer data to the energy cloud server as the input data for better processing and less energy consumption.

Software Model

In this subsection, we design a software model of an IoT-based energy management system with intelligent edge computing, as shown in Fig. 2. The proposed software model includes four layers: the sensing layer, network layer, cognition layer, and application layer.

Sensing Layer: In the sensing layer, the devices can generate or detect the energy information of the connected energy network. The energy edge server is responsible for managing connection among devices, that is, setting up reliable communication links for devices. Querying data is one of the basic features of the proposed software model required by intelligent edge computing services. Considering the heterogeneity of the energy data in smart cities, the edge server will queue and classify the collected energy data for hierarchical processing.

Network Layer: In the proposed IoT-based energy management system, data transmission is a crucial function for data transferring between energy devices and the energy edge server, and task offloading between the edge server and the cloud server. Different kinds of communication technologies, such as power line communications (PLC), 5G, LTE, and WiFi, can be used for data transmission. The data storage capacity in the energy cloud server, energy edge, and energy devices can be integrated as one source of network storage, and the energy data stored in such storage can generate a "data pool." By using a unified interface, heterogeneous data is able to be accessed from the data pool by any devices

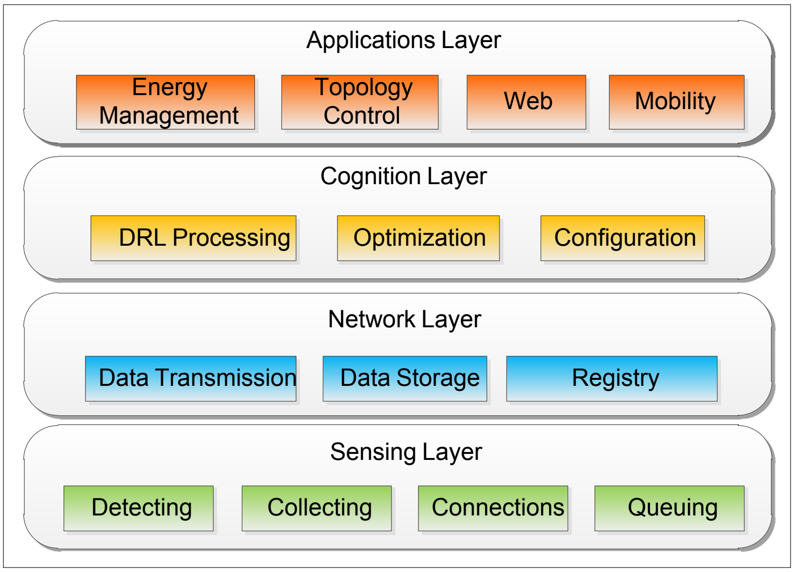

FIGURE 2. The software model of an IoT-based energy management system.

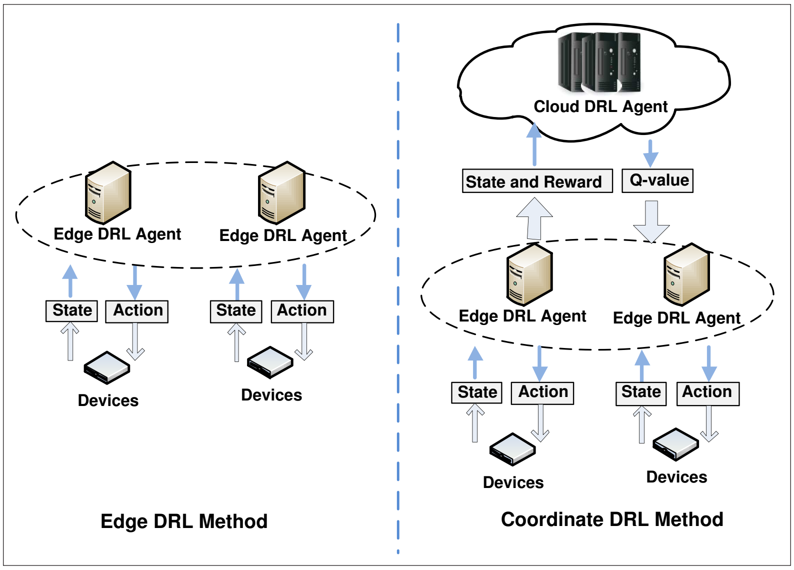

FIGURE 3. The DRL process.

and servers. The virtual data pool can help energy managers in developing control policies by historical data analysis. The registry is used to record the dynamic entering/leaving of devices in the proposed IoT-based energy network. Since the devices may enter/leave the network frequently, the registry plays a significant role in supporting related network configuration.

Cognition Layer: In the proposed system structure, the cognition layer is a core layer that creates intelligent awareness of the energy environment. This layer contains three main functional modules: DRL processing, optimization, and configuration. DRL processing modules are deployed at the cloud server and edge server. The module records the demands and current states of associated users. Then, according to the results of the last executed action, the reward can be calculated by the DRL module. The DRL module can make decisions under the guidance of a powerful DNN that provides accurate estimation and prediction. Optimization function is very important for a DRL agent to obtain an optimal solution in online deep Q-learning for entire operational periods. Since the frequent usage of edge servers may cause unacceptable energy consumption, the optimization also can explore an optimal schedule of using edge servers for cost minimization. The configuration can be realized in the energy cloud server to configure the operations of edge servers or devices. Meanwhile, the configuration also can be operated in the edge server for nearby devices. Note that the configuration can be processed at the cloud server in a centralized manner or at each device/edge server in a decentralized manner.

Application Layer: The application layer provides a set of functions and tools to postprocess data coming from the underlying layers and figurie out the network settings of the proposed IoT-based infrastructure. In particular, energy management is the core function to help the entities control and schedule the energy from every angle of the system without knowing the underlying layer conditions. Topology control is used to make the decision for devices leaving/entering the network. Web-based application (e.g., online dashboard for visual management) and mobility application (e.g., mobile applications on smartphones for facility managers) benefiting the interoperability among different devices and technologies is enabled in the proposed IoT-based energy management system.

Deep Reinforcement Learning for Efficient Energy Scheduling

In this section, we present a DRL-based processing framework (Fig. 3), in which the DRL agents are deployed at cloud and edge servers, respectively. Two types of DRL methods are proposed: the edge DRL method and the cooperative DRL method. In the first method, the devices offload the energy scheduling task to the edge server. Then the edge server uses the DRL method to find optimal scheduling results for devices. In the second method, the edge server will offload the DNN training to the cloud server to reduce the computation cost and then adopt the deep Q-learning process based on the estimated Q-value from the cloud server.

Edge DRL Method

Offline DNN Construction with Devices Classification: For the DNN construction, we employ a feed-forward neural network that has two hidden layers of fully connected softplus units with 64 and 32 neurons, respectively. In order to train the DNN, we use the same training method as [15]. Then we define the state space, action space, and reward function of the cloud DRL framework as follows.

State Space: The state of the edge DRL agent consists of the demands $R_u \in [R_{min}, R_{max}]$ of each user u. The state vector is represented as $[R_1 \ldots, R_U]$ with a cardinality of U.

Action Space: To satisfy different service requirements, we let the edge DRL agent divide the devices into different classes in each decision epoch. Specifically, the DRL agent determines which class of devices should be served in the current epoch. After executing an action, the DRL agent can derive an active set of a class of devices.

Reward: The reward is used to achieve the minimal energy consumption for the devices and satisfy devices' demands. We define the immediate reward that the cloud DRL agent receives as $\mathbb{R}_d = e_{max} - e_{real}$, where e_{max} is the maximum possible value of the energy consumption of the devices and e_{real} gives the actual total energy consumption of the devices that can be obtained at each decision epoch.

Online Deep Q-Learning: After the offline DNN tranning, the deep Q-learning is employed and can be described as follows:

- *Step 1:* At the beginning of each decision epoch, the edge DRL agent derives the estimated Q function value by the DNN with the input of state-action pair for each state from devices.
- *Step 2:* For a specific class of devices, the ε-greedy policy is used for selecting execution action. In this policy, the action is chosen according to the highest estimated Q-value with (1 – ε) probability. With probability e, an action is randomly selected from the action set.
- *Step 3:* The edge DRL agent obtains the optimal solution based on the selected class of devices and manages the energy usage of devices with the optimal solution.
- *Step 4:* The state transition information of the class of devices is stored in the experience memory and sampled after observing the immediate reward and next state from the devices.
- *Step 5:* At the end of the decision epoch, the sampled state transitions are used to obtain a loss function for the edge DRL agent to update the DNN's weights.

Cooperative DRL Method

Offline DNN Construction in Cloud: In this method, the offline DNN training is send to the cloud server by considering the capacity limitation of the edge server. The cloud DRL agent will execute the DNN construction with same training method shown above. The state space, action space, and reward are described as follows:

State Space: The state of DRL agent consists of the demands $R_{i,u} \in [R_{min}, R_{max}]$ of each user u from edge server i. The state vector is represented as $[R_{1,1} \dots , R_{N,U}]$ with a cardinality of $(N + U)$.

Action Space: To satisfy different computation requirements, we let the cloud DRL agent make the priority decision of the edge servers in each decision epoch. Specifically, the DRL agent determines which edge server to be served or not. After executing an action, the DRL agent can derive an active set of edge servers.

Reward: The reward is used to achieve the minimal energy consumption for the entire energy management system and satisfying devices' demands. We define the immediate reward that the DRL agent receives as $\mathbb{R}_e = E_{max} - E_{real}$, where E_{max} is the maximum possible value of total energy consumption and E_{real} gives the actual total energy consumption that can be obtained at each decision epoch.

Online Deep Q-Learning on Edge Server: After the offline DNN tranning, the deep Q-learning is employed at the edge DRL agent for online dynamic control. The process of online deep Q-learning on the edge server can be described as follows:

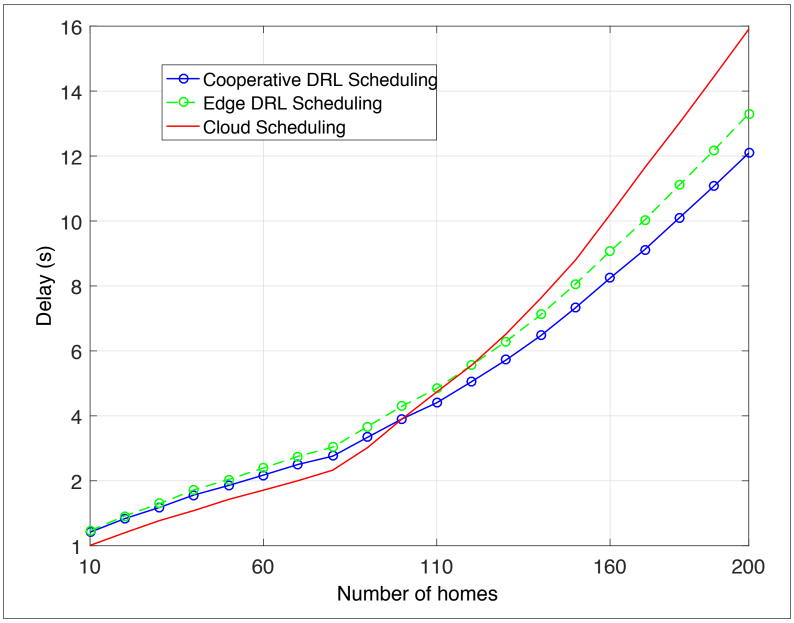

FIGURE 4. Energy cost comparison among different energy scheduling schemes.

- *Step 1:* At the beginning of each decision epoch, the edge DRL agent derives the estimated Q-value by the DNN from the cloud DRL agent with the input of state-action pair for each state.
- *Step 2:* To select the execution action, the ε-greedy policy is still used.
- *Step 3:* After observing the immediate reward and next state from the devices, the state transition information is stored in the experience memory in the edge server.
- *Step 4:* At the end of the decision epoch, the edge DRL agent updates the parameters of the DNN with given samples from the experience memory.

Illustrative Results

In this section, we consider 5 communities, each of which deploys a edge computing server and consists of 40 homes and 10 renewable resources. The RELOAD database, [16] which provides hourly load profiles of different practical demands including HVAC, water heating, lighting, clothes drying, freezing, and so on, is used to model different demands of homes. The market prices for purchasing electricity from the grid is $p(t) = 0.3$ cents in daytime hours, that is, from 8:00 a.m. to 12:00 a.m., and $p(t) = 0.2$ cents during the night, that is, from 12:00 a.m. to 8:00 a.m. the day after. Two metrics are considered in the simulation: energy cost and delay. Moreover, we compared the two proposed DRL-based energy scheduling schemes (denoted as "cooperative DRL scheduling" and "edge DRL scheduling") with a baseline method named "cloud scheduling" in which the cloud server directly provides the energy scheduling for homes without DRL method.

Energy Cost

The energy cost is defined as the average cumulative cost of the homes during a day.

The energy cost comparison of cooperative DRL scheduling, edge DRL scheduling, and cloud scheduling schemes are shown in Fig. 4.

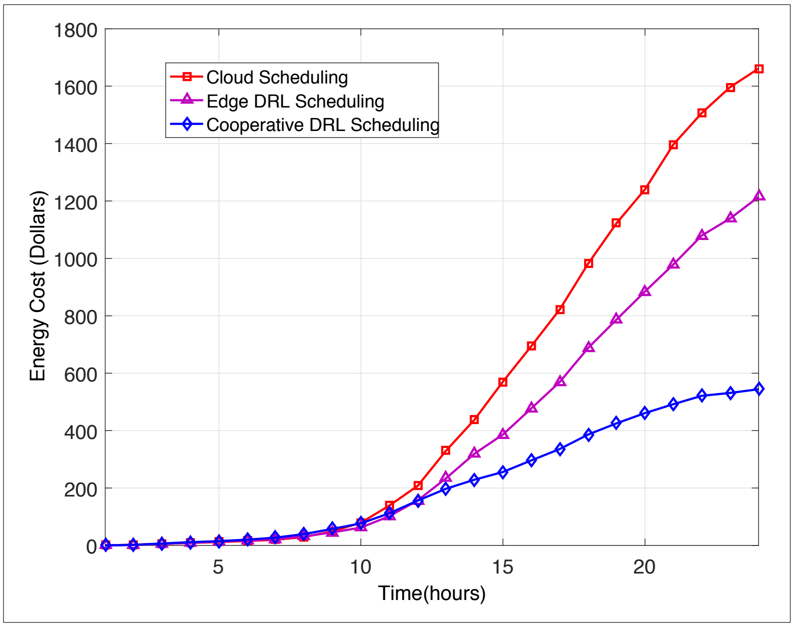

FIGURE 5. Delay comparison among different energy scheduling schemes.

It is observed that the energy costs achieved by the proposed DRL-based schemes are lower than those of the baseline scheme. This is because the proposed DRL-based scheme can find the optimal energy scheduling policy by predicting the profiles of the dynamic demands and renewables. The baseline method may may be significantly affected by the randomness of the users' demands and renewables in reality.

It is also noted that the energy cost in the edge DRL scheduling scheme is higher than that of the cooperative DRL scheduling scheme. That is, without the help of a cloud server, the edge server should operate the offline training and deep Q-learning for the homes by itself. Considering the capacity limit of the edge server, the scheduling requirements of some homes will be hung up, resulting in extra energy costs.

Delay

The delay is defined as the the execution time of the tasks from each home plus the time for communication between devices and the upper layer server (including the cloud server and edge computing server).

Figure 5 shows the delay comparison of the cooperative DRL scheduling edge DRL scheduling, and cloud scheduling schemes. It is expected that the delay of these three schemes increases as the number of homes increase. It is interesting to note that the cloud scheduling scheme first has lower delay and then has higher delay than that of the other two proposed DRL-based schemes as the number of homes increase. That is, in the cloud scheduling scheme, the homes will send the computation requirements to the cloud server at the same time, which may cause transmission congestion. Therefore, the transmission delay will be drastically increased with the increment of the number of homes.

In the edge DRL scheduling scheme, the homes in different classes have different priorities to send the requirements, which means the homes will not compete for transmission opportunities at the same time and result in lower delay compared to cloud scheduling. However, this scheme still cause the execution delay since the edge server may not be able to handle numerous devices' tasks in a timely manner with limited computation capacity. In the cooperative DRL scheduling scheme, easy parts (Q-learning) of the scheduling task can be executed at the edge server near the homes, and the complex parts (DNN training) can be sent to the cloud for execution. This scheme can reduce both execution and transmission delay as the number of homes becomes large.

Conclusion

In this article, the IoT-based energy management framework for a smart city is surveyed and studied. By introducing the software model of the proposed framework, the promotion of enabling edge computing technology is introduced. Then, to deal with the intermittence and uncertainty of the energy supplies and demands in cities, a DRL-based energy scheduling scheme is presented for a long-term goal. The efficiency of the energy scheduling scheme is analyzed under the cases with and without edge servers, respectively. In illustrative results, we can observe that the proposed schemes can achieve low energy cost while causing lower delay compared to traditional schemes.

Acknowledgment

The corresponding author is Shengli Xie (shlxie@gdut.edu.cn). The work is supported in part by programs of NSFC under grant nos. 61773126, 61603099, 61701125, Pearl River S&T Nova Program of Guangzhou, No. 201806010176. The European Unions Horizon 2020 research and innovation programme under the Marie Skodowska-Curie grant agreement No. 824019.

References

[1] WHO, "Urban Population Growth"; http://www.who.int/gho/urban health/situation trends/urban population growth text/en/,accessed Feb. 2016.
[2] D. V. Gibson *et al*.,, *The Technopolis Phenomenon, Smart Cities, Fast Systems, Global Networks*, Lanham, Md. Rowman & Littlefield Publishers, 1992.
[3] A. Caragliu *et al*., "Smart Cities in Europe," *Ideas*, vol. 98, no. 5, 2009, pp. 1766–82.
[4] L. Da Xu *et al*., "Internet of Things in Industries: A Survey," *IEEE Trans. Industrial Informatics*, vol. 10, no. 4, 2014, pp. 223–43.
[5] Y. Liu *et al*., "SD-MAC: Spectrum Database-Driven MAC Protocol for Cognitive Machine-to-Machine Networks," *IEEE Trans. Vehic. Tech.*, vol. 66, no. 2, Feb. 2017, pp. 1456–67.
[6] H. Li *et al*., " Deep Reinforcement Learning: Framework, Applications, and Embedded Implementations," invited paper, *2017 IEEE/ACM Int'l. Conf. Computer-Aided Design*, 2017, pp. 847–54.
[7] J. Ren *et al*., "Serving at the Edge: A Scalable IoT Architecture Based on Transparent Computing," *IEEE Network*, vol. 31, no. 5, Sept./Oct. 2017, pp. 96–105.
[8] H. Li *et al*., " Learning IoT in Edge: Deep Learning for the Internet of Things with Edge Computing," *IEEE Network*, vol. 32, no. 1, Jan./Feb. 2018, pp. 96–101.
[9] D. Snoonian, "Smart Buildings," *IEEE Spectrum*, vol. 40, no. 8, 2003, pp. 18–23.
[10] Z. Xu *et al*.,"Performance Analysis and Comparison on Energy Storage Devices for Smart Building Energy Management," *IEEE Trans. Smart Grid*, vol. 3, no. 4, 2012, pp. 2136–47.
[11] M. Masera *et al*.,"Smart (Electricity) Grids for Smart Cities: Assessing Roles and Societal Impacts," *IEEE Proc.*, vol. 99, no. 6, Apr. 2018, pp. 613–25.
[12] D. A, "Modelling Socio-Economic and Energy Aspects of Urban Systems," *Sustainable Cities and Society*, vol. 13, 2014, pp. 192–206.

[13] Y. Liu *et al.*, "Electricity Cost Minimization for a Microgrid with Distributed Energy Resource under Different Information Availability," *IEEE Trans. Industrial Electronics*, vol. 62, no. 4, Apr. 2015, pp. 2571–83.
[14] W. Zhong *et al.*, "Auction Mechanisms for Energy Trading in Multi- Energy Systems," *IEEE Trans. Industrial Informatics*, vol. 14, no. 4, 2018, pp. 1511–21.
[15] V. Mnih *et al.*, "Human-Level Control Through Deep Reinforcement Learning," *Nature*, vol. 518, no. 7540, 2015, pp. 529–33.
[16] N. Hassan *et al.*, "Impact of Scheduling Flexibility on Demand Profile Flatness and User Inconvenience in Residential Smart Grid," *Energies*, vol. 6, no. 12, Dec. 2013, pp. 6608–35.

Biographies

Yi Liu (yi.liu@gdut.edu.cn) received his Ph.D. degree from South China University of Technology (SCUT), Guangzhou, in 2011. After that, he joined the Singapore University of Technology and Design (SUTD) as a postdoctoral researcher. In 2014, he worked in the Institute of Intelligent Information Processing at Guangdong University of Technology (GDUT), where he is now a full professor.His research interests include wireless communication networks, cooperative communications, smart grid, and intelligent edge computing.

Chao Yang (yangchaoscut@aliyun.com) received his Ph.D. degree in signal and information processing from SCUT in 2013. He is currently with the School of Automation, GDUT. From 2014 to 2016, he was a research associate with the Department of Computing, Hong Kong Polytechnic University. His research interests focus on VANETs, smart grid, and edge computing.

Li Jiang (jiangli@gdut.edu.cn) received her Ph.D. degree from the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications (BUPT), China, in 2017. She is a lecturer in the School of Automation, GDUT. Her current research interests include 5G networks and beyond, resource management, physical layer security, and blockchain.

Shengli Xie [M'01, SM'02, F'19] (shlxie@gdut.edu.cn) received his M.S. degree in mathematics from Central China Normal University, Wuhan, in 1992, and his Ph.D. degree in automatic control from SCUT in 1997. He was a vice dean with the School of Electronics and Information Engineering, SCUT from 2006 to 2010. Currently, he is the director of both the Institute of Intelligent Information Processing (LI2P) and Guangdong Key Laboratory of Information Technology for the Internet of Things, and also a professor with the School of Automation, GDUT. He has authored or co-authored four monographs and more than 100 scientific papers published in journals and conference proceedings, and has been granted more than 30 patents. His research interests broadly include statistical signal processing and wireless communications, with an emphasis on blind signal processing and the Internet of Things.

Yan Zhang [SM'10] (yanzhang@ieee.org) is a full professor at the University of Oslo. He is an Editor of several IEEE publications, including *IEEE Communications Magazine*, *IEEE Network*, *IEEE Transactions on Green Communications and Networking*, *IEEE Communications Surveys & Tutorials*, and the *IEEE Internet of Things Journal*. His current research interests include next generation wireless networks leading to 5G and cyber physical systems. He is an IEEE VTS Distinguished Lecturer and a Fellow of IET. He received the "Highly Cited Researcher" award (Web of Science top 1 percent most cited worldwide) according to Clarivate Analytics.