Adaptive Edge Sensing for Industrial IoT Systems: Estimation Task Offloading and Sensor Scheduling

Ling Lyu, *Member, IEEE*, Lihong Zhao, Yanpeng Dai, *Member, IEEE*, Nan Cheng, *Member, IEEE*, Cailian Chen, *Member, IEEE*, Xinping Guan, *Fellow, IEEE*, and Xuemin Shen, *Fellow, IEEE*

***Abstract*—Edge sensing can achieve high-performance state estimation in industrial IoT systems by supporting task offloading and data processing at powerful edge estimators. Accurate edge sensing depends on low offloading delay. However, it is challenging to decrease offloading delay due to the harsh industrial environment and limited communication-and-computation resources. In this article, a closed-form expressing of estimation error with respect to offloading delay is derived to indicate that adjusting offload delay on demand is necessary for estimation error reduction. Then, we propose an adaptive edge sensing scheme, aiming to minimize estimation error by jointly optimizing task offloading and sensor scheduling. The required optimization is formulated as a mixed-integer nonlinear programming problem and solved by the designed decomposition and approximation methods. Specifically, the maximum matching is used for sensor scheduling to assign the optimal edge estimator for each sensor. The task offloading algorithm is designed based on the inner approximation method to reduce the offloading delay. Finally, simulation results demonstrate that the proposed scheme has superiorities in reducing estimation error compared with centralized sensing and distributed sensing schemes. Moreover, we find an interesting result that estimation error is delay sensitive when the offloading delay is large.**

***Index Terms*—Edge sensing, industrial IoT systems, offloading delay, sensor scheduling, task offloading.**

Manuscript received 24 March 2022; revised 10 July 2022; accepted 15 August 2022. Date of publication 22 August 2022; date of current version 22 December 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 62002042, Grant 62071356, Grant 62101089, Grant 92167205, and Grant 61933009; in part by the China Postdoctoral Science Foundation under Grant 2021M690022 and Grant 2021M700655; in part by the Fundamental Research Funds for the Central Universities under Grant JB210113; and in part by the Cooperative Scientific Research Project, Chunhui Program of Ministry of Education, China. *(Corresponding author: Yanpeng Dai.)*

Ling Lyu is with the School of Information Science and Technology, Dalian Maritime University, Dalian 116026, China, and also with the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China (e-mail: linglyu@dlmu.edu.cn).

Lihong Zhao and Yanpeng Dai are with the School of Information Science and Technology, Dalian Maritime University, Dalian 116026, China (e-mail: lihongzhao@dlmu.edu.cn; yanpengdai@dlmu.edu.cn).

Nan Cheng is with the School of Telecommunications Engineering, Xidian University, Xi'an 710071, China (e-mail: nancheng@xidian.edu.cn).

Cailian Chen and Xinping Guan are with the Department of Automation and the Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: cailianchen@sjtu.edu.cn; xpguan@sjtu.edu.cn).

Xuemin Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: sshen@uwaterloo.ca).

Digital Object Identifier 10.1109/JIOT.2022.3200392

I. Introduction

With the rapid deployment of information and communications technologies, ubiquitous sensing over Internet of Things (IoT) plays an increasingly important role in industrial automation [1], [2], [3]. In industrial IoT systems, ubiquitous sensing is archived by performing state estimation with spatially distributed sensors and multiple estimators. In particular, sensors cooperatively sense the control system states, and then deliver the sensory data to estimators. Based on the received sensory data, estimators performs state estimation to estimate the system state parameters [4]. In general, the more sensory data delivered to estimators, the more accurate the state estimation can be. However, due to limited communication resources in IoT networks, sensory data may be discarded in the case of congestion [5], [6], [7], [8], which, in turn, deteriorates estimation accuracy and wastes communication resources. Moreover, the serious fading and complex interference in the harsh industrial environment, may lead to enlarging transmission delay and even transmission failure of sensory data.

For the issue of how to design state estimation algorithms to alleviate the impact of wireless transmissions, there has been significant theoretical advancement in the area of networked control systems. On one hand, some works focus on designing the robust state estimation algorithms [9], [10]. Li *et al.* [11] designed a finite-horizon estimator to work against the issue of channel fading. Xu *et al.* [12] proposed a local coupling structure based state estimators to address the issue of robust state estimation for coupled neural networks with randomly occurring distributed delays. These works pay attentions on the worse case analysis to against the adverse effects introduced by wireless networks, such as random packets loss, random delay, and limited resources. However, the wireless communication quality in practice is generally better than the worse case, which makes the algorithm design and performance analysis of state estimation too conservative. On the other hand, some works pay efforts on investigating resource-effective state estimation algorithm to reduce the demand of network resources, such as the distributed state estimation [13]. Rana *et al.* [14] proposed a distributed consensus estimation algorithm, where local estimates are exchanged via wired networks. Zhang *et al.* [15] presented a distributed fusion estimation method to reduce communication costs for the control system observed by wireless sensor networks with given packet losses. However, the limited communication resources and harsh industrial environment (such as the serious

fading, complex interference, and path loss) may render the transmission delay and pack loss of local estimates unpredictably and uncontrollably, and then degrade the consensus performance and estimation accuracy.

For the issue of limited network resources, one solution is to employ the edge computing technology in the area of wireless communications [16]. Recently, the resource allocation problem in edge computing enabled wireless communications has attracted significant attentions [17], [18], [19], [20]. Dai *et al.* [21] jointly optimized the computation offloading and the user association to minimize the overall energy consumption. Ye *et al.* [22] jointly maximized the communication and computing resource utilization with diverse quality-of-service guarantee for autonomous driving tasks. Zhang *et al.* [23] jointly optimized computation offloading and resource allocation to balance the energy consumption and execution latency. Moreover, the content caching, computation offloading, spectrum, and computation resource allocation were jointly considered to improve the performance of wireless cellular networks [24]. Considering the energy consumption of both the task computing and task data transmission, Kuang *et al.* [25] investigated the joint problem of partial offloading scheduling and resource allocation. Inspired by this, we focus on the edge computing enabled state estimation, where the contradiction between massive sensory data and the limited communication resources could be relieved.

Recent advances on hardware accelerators enables substantial processing to be performed at sensors locally, thus the task of state estimation could be executed at end sensors or edge estimators. This raises a new question that which device performs the estimation tasks will be better for estimation accuracy improvement under the limitation of available communication and computation capacity. Specifically, if the state estimation is performed at end sensors, the processed local estimates are delivered to edge estimators. In this way, the local processing typically reduces delivered data volume, but it usually entails a larger computational delay due to the weaker computational capability of sensors. More important, if the local estimates is delayed or lost, the estimation error will be significantly deteriorated since no raw sensory data could be used to infer the current state. On the contrary, if the estimation task is executed by edge estimators, the raw sensory data are needed to be delivered. As the data volume of raw sensory data is much larger than that of local estimate, it will incur a larger communication delay even though it has advantages on reducing the computational delay. Note that if only a few of raw sensory data are delayed or lost, the estimation error could not increase dramatically.

In this article, we design an adaptive end sensing scheme for industrial IoT systems, where each sensor makes the offloading decision according to the accuracy demand of the estimation task and the offloading delay of sensory data. The estimation error is then minimized by jointly optimizing task offloading and sensor scheduling with limited communication and computation resources. The contributions of this article are summarized as follows.

1) The impact of the experienced delay on the state estimation accuracy is explicitly characterized, which indicates that the mean square error of state estimation exponentially increases with the growth of experienced delay of sensory data. The relationship between estimation error and experienced delay is the cornerstone of designing adaptive edge sensing (AES) for industrial IoT systems.
2) The designed AES approach has superiorities in reducing estimation error, since it makes the offloading decision by considering not only the communication and computational resources but also the accuracy demand required by state estimation and the observation ability of sensors.
3) The near-optimal solution of formulated mixed-integer nonlinear programming problem is obtained by estimation error analysis, objective function approximation, and optimization problem decomposition. Specifically, the maximum matching is used for sensor scheduling to assign the optimal edge estimator for each sensor. The task offloading algorithm is designed based on the inner approximation method to reduce the offloading delay.

The remainder of this article is organized as follows. The system and communication models are presented in Section II. In Section III, an adaptive end sensing is designed for resource-limited industrial IoT systems. In Section IV, the estimation task offloading and sensor scheduling are jointly optimized to further enhance the state estimation accuracy. Simulation results and main conclusions are shown in Sections V and VI, respectively.

II. System Models

As shown in Fig. 1, the considered industrial IoT system is composed of N sensors, N edge estimators and one remote coordinator, where the edge estimators can only coverage a part of sensors. Moreover, one sensor could be in the coverage regions of different edge estimators. In the considered industrial IoT systems, all sensors deliver the collected measurements or the calculated local estimates to edge devices. The data volume of sensing information transmitted by sensors are significantly larger than that of fusion estimates exchanged among edge devices. Moreover, edge devices perform fusion estimation based on the received sensing information, and then exchange the fusion estimates with each other. Compared with the raw measurements and the calculated local estimates, the fusion estimates carry more information and are more sensitive to transmission delay and packet loss. Therefore, all edge estimators are connected with the wired network. We consider that all edge estimators have the processing and computing ability and, thus, they take charge of estimating the system state with the received raw sensory data from sensors. In this work, we consider that sensors not only has the capacity of sensing and transmitting but also can process the sensory data and perform the local state estimation.

A. System Model

The work considers the distributed fusion estimation for the networked multisensor fusion systems, where all sensors are assumed to be synchronized and have the same measurement rate [26], [27]. The considered discrete-time linear

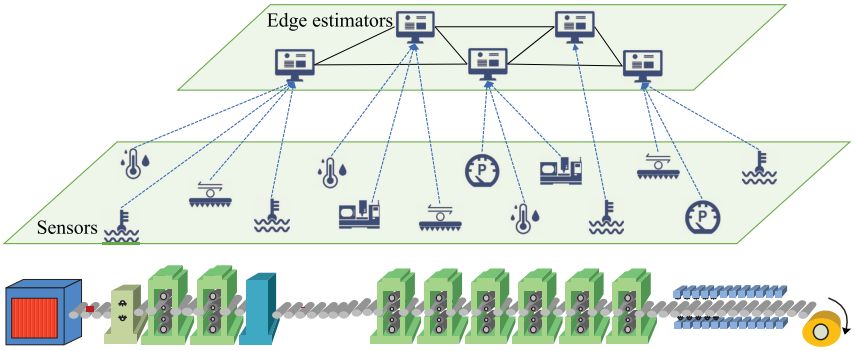

Fig. 1. Application of the industrial IoT system in hot rolling process.

time-invariant control system with N sensors is described by

$$x(t+1) = Ax(t) + Fw(t) \tag{1}$$

$$y_i(t) = C_i x_i(t) + v_i(t), i = 1, 2, \ldots, N \tag{2}$$

where $x(t) \in R^n$ is the state of the process, $y_i(t) \in R^{m_i}$ is the measure of the ith sensor, and $w(t) \in R^q$ and $v_i(t) \in R^{m_i}$ are the zero-mean white noises with covariances $Q_w(t)$ and $Q_{v_i}(t)$, respectively. The initial state $x(0)$, the input noise $w(t) \in R^q$, and the measurement noise $v_i(t) \in R^{m_i}$ are mutually uncorrelated. The matrices $A \in R^{n\times n}$, $F \in R^{n\times q}$, and $C_i \in R^{m_i\times n}$ are time invariant. In addition, the duration of one time step is unitized.

B. Communication Model

In this work, we consider that all edge estimators share the same resource block. In order to avoid the conflict among edge estimators, the edge estimator can schedule only one sensor to transmit sensory data on one resource block. Let the binary variable $\delta \in \{1, 0\}$ denote the sensor scheduling indicator. Particularly, if the ith sensor is scheduled by the sth edge estimator at the tth time step, $\delta_{i,s}(t) = 1$. Otherwise, $\delta_{i,s}(t) = 0$ $(i = 1, 2, \ldots, M)$. The power channel gain between the ith sensor and the sth edge estimator at the tth estimation step is denoted as $g_{i,s}(t)$ $(s = 1, 2, \ldots, N)$. In this article, the channel gains are assumed to be known, which could be obtained with channel estimation or an empirical channel model. In the considered industrial IoT system, the edge estimators in general are much closed to sensors than the remote coordinator, resulting the better power channel gain. However, due to the shelter of equipments in the filed, the power channel gain between the sensor and the nearby edge estimators are worse, as shown in Fig 2. Without loss of generality, some sensors may be not scheduled by any edge estimators. In this manuscript, the frequency-division duplex mode is used. But, the proposed scheme is also suitable in the time-division duplex mode. In particular, in the frequency-division duplex mode, multiple sensors transmit data on different channels simultaneously. In the time-division duplex mode, the duration of one estimation step could be divided into multiple slots, and different sensors transmit data on different slots.

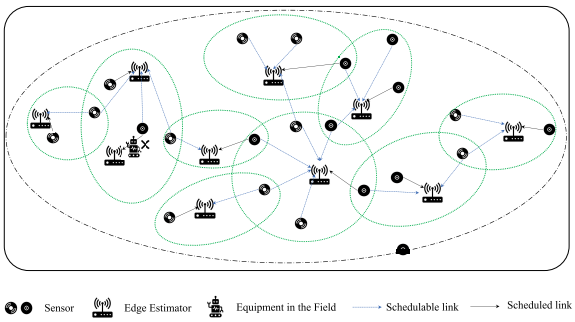

Fig. 2. Network architecture for the considered industrial IoT system.

III. Adaptive Edge Sensing

Due to the lossy wireless channels, the sensory data and local estimates may be lost or delayed, which will deteriorate the accuracy of state estimation. In this section, we first discuss the experienced delay, and then analyze the impact of experienced delay on the estimation accuracy. Finally, sensor scheduling and task offloading are jointly designed to minimize the mean square error of state estimation.

A. Experienced Delay

In the considered IoT system, the tasks generated by different sensors are independent. Let $K_i(l_i, l_i^e, m_i)$ denote the task generated by the ith sensor, where l_i is the data size of sensory data, l_i is the data size of local estimate, and m_i is the number of the required CPU-cycles to process the received sensory data. In addition, the computing speed of the ith sensor and the sth edge estimator is denoted as f_i^l and f_s^e (cycles/s), respectively. Then, the computing delay for the ith sensor to process its own sensory data locally is

$$\tau_i^l = \frac{m_i}{f_i^l}. \tag{3}$$

Similarly, the computing delay for the sth edge estimator to process the task of the ith sensor is

$$\tau_{i,s}^e = \frac{m_i}{f_s^e}. \tag{4}$$

For each scheduled sensor, the experienced delay of sensory data is composed of the transmission delay and the computing

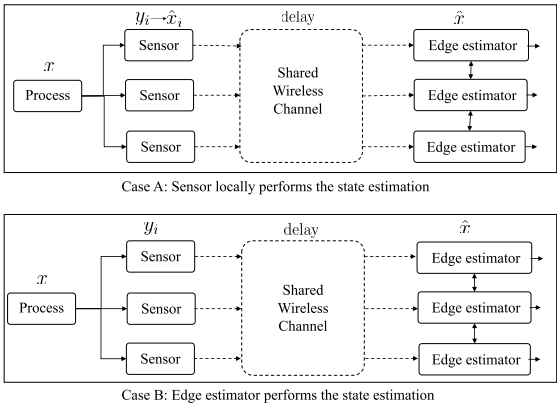

Fig. 3. Two cases of state estimation.

delay. The achievable data rate of the scheduled link from the sensor to the edge estimator is given by

$$r_{i,s}(t) = \log\left(1 + \frac{p_i(t)g_{i,s}(t)}{\sum_{i' \neq i, i' \in \mathcal{N}} p_{i'}(t)g_{i',s}(t) + N_0}\right) \tag{5}$$

where $p_i(t)$ is the transmit power of the ith sensor at the tth time step. Due to the battery-powered sensor, the transmit power should satisfy that $p_i(t) \leq p_{\max}$, where $p_{\max}$ is the maximum transmit power of each sensor. At each time step t, the ith sensor can only be scheduled by at most of one edge estimator, thus $\sum_{s\in\mathcal{S}} \delta_{i,s}(t) \leq 1$. If the sensor is not scheduled by any estimators, the experienced delay is set to be $\mathcal{T}$, where $\mathcal{T}$ the tolerated maximum delay. In this regard, the transmission delay of sensory data and local estimate is given by

$$\tau^{lt}_{i,s}(t) = \frac{l_i}{r_{i,s}(t)} \tag{6}$$

$$\tau^{et}_{i,s}(t) = \frac{l^e_i}{r_{i,s}(t)}. \tag{7}$$

In summary, the experienced delay is the integration of computing delay and transmission delay, shown as follows:

$$\tau_{i,s}(t) = \theta_i(t)\left(\tau^e_{i,s} + \tau^{et}_{i,s}(t)\right) + (1-\theta_i(t))\left(\tau^l_i + \tau^{lt}_{i,s}(t)\right) \tag{8}$$

where the binary variable $\theta_i(t)$ is the task offloading decision indicator. If $\theta_i(t) = 1$, the ith sensor directly delivers the sensory data to the assigned edge estimator at the tth time step. Otherwise, the ith sensor processes the sensory data locally and then delivers the calculated local estimate to the assigned edge estimator. For notation simplicity, let $\tau_i(t) = \sum_{s\in\mathcal{S}} \delta_{i,s}(t)\ \tau_{i,s}(t)$. If $\tau_i(t) > \mathcal{T}$, the data packet generated by the ith sensor is considered to be lost, and the experienced delay is set to be $\mathcal{T}$.

B. Edge Sensing

In the considered industrial IoT system, each sensor has to determine whether performs the local estimation or not. Therefore, there are two cases of state estimation, shown in Fig. 3. The offloading decision will affect the experienced delay of sensing information, which plays an important role in the estimation accuracy. The sensing information with a large experienced delay will enlarge the estimation error, since the sensing information used to perform the state estimation is out of fashion. In case A, each sensor executes the local estimation based on its own sensory data, and then delivers the local estimate to the edge estimator over shared wireless channels. The edge estimator fuses the local estimates received from the scheduled sensors and other edge estimators. In case B, the sensory data are directly delivered to edge estimators, and the edge estimator combines all received sensory data to obtain the edge estimate. The calculated edge estimate is fused with other edge estimates. The detailed state estimation process is elaborated as follows.

1) Case A (Sensor Locally Performs the State Estimation): Based on all sensory data $\{y_i(1), y_i(2), \ldots, y_i(t)\}$, the ith sensor could obtain the local optimal estimate in terms of linear minimum variance sense by recursively calculating the standard Kalman filter [28]

$$\hat{x}_i(t) = [I_n - K_i(t)C_i(t)]A_i(t-1)\hat{x}_i(t-1) + K_i(t)y_i(t)$$
$$K_i(t) = P^-_{ii}(t)C_i(t)^T\left[C_i(t)P^-_{ii}C_i(t)^T + Q_{V_i}(t)\right]^{-1}$$

where $P^-_{ii}(t)$ is the one-step prediction error covariance matrix. Then, the estimation error covariance matrix $P_{ii}(t) = \mathbb{E}\{[x_i(t) - \hat{x}_i(t)][x_i(t) - \hat{x}_i(t)]^T\}$ is expressed as

$$P_{ii}(t) = [I_n - K_i(t)C_i(t)]P^-_{ii}(t)$$
$$P^-_{ii}(t) = A_i(t-1)P_{ii}(t-1)A_i(t-1)^T + F(t-1)Q_w(t-1)F(t-1)^T.$$

The estimation error cross-covariance matrix $P_{ij}(t) = \mathbb{E}\{[x_i(t) - \hat{x}_i(t)][x_i(t) - \hat{x}_j(t)]^T\}$ is expressed as

$$P_{ij}(t) = [I_n - K_i(t)C_i(t)]\left[A_i(t-1)W_{ij}(t-1)A_i(t-1)^T + F(t-1)Q_w(t-1)F(t-1)^T\right]\left[I_n - K_j(t)C_j(t)\right]^T (i \neq j).$$

Considering the experienced delay, the edge estimate obtained by the sth edge estimator is denoted as $\hat{x}^e_i(t|t)$ ($\delta_{i,s} = 1$). The edge estimator will store the most recent edge estimate and discards older edge estimates. At the tth time step, if the stored edge estimate is $\hat{x}^e_i(t|t-\tau_i)$

$$\hat{x}^e_i(t|t-\tau_i) = \left[\prod_{h=1}^{\tau_i} A_i(t-h)\right]\hat{x}_i(t-\tau_i|t-\tau_i) \tag{9}$$

which indicates that the estimation error exponentially increases with the growth of offloading delay. The fusion estimate is $\hat{x}(t|t-\tau_i) = \sum_{i=1}^N \Omega_i(t)\hat{x}^e_i(t|t-\tau_i)$, where the optimal weighted matrices $\{\Omega_1(t), \ldots, \Omega_N(t)\}$ are given by

$$[\Omega_1(t), \ldots, \Omega_N(t)] = \left[I_0^T\Xi^{-1}(t)I_0\right]^{-1}I_0^T\Xi^{-1}(t) \tag{10}$$

where $\Xi(t) = \mathbb{E}\{[e_1^T(t-\tau_1), \ldots, e_N^T(t-\tau_N)]^T[e_1^T(t-\tau_1), \ldots, e_N^T(t-\tau_N)]\}$ with $e_i(t-\tau_i) = x(t) - \hat{x}^e_i(t|t-\tau_i)$, and $I_0 = [I_n, \ldots, I_n]$. As $\sum_{i=1}^N \Omega_i(t) = I_n$, the fusion estimation is unbiased when $\mathbb{E}\{\hat{x}^e_i(t|t-\tau_i)\} = \mathbb{E}\{x(t)\}$. Furthermore, the error covariance matrix of fusion estimation is given by

$$\mathbb{E}\left\{\left(x(t) - \hat{x}(t)\right)\left(x(t) - \hat{x}(t)\right)^T\right\} = \left[I_0^T\Xi^{-1}(t)I_0\right]^{-1}. \tag{11}$$

In case A, the impact of experienced delay reflects on the edge estimate.

2) Case B (Edge Estimator Performs the State Estimation): The sensory data received by at the sth edge estimator (when the ith sensor is scheduled to access to the sth edge estimator) is $\{\beta_i(1)y_i(t-(t-1)), \ldots, \beta_i(t-\tau_i)y_i(t-\tau_i), \ldots, \beta_i(t)y_i(t)\}$, where the binary variable $\beta_i(t) \in \{0, 1\}$ is introduced to indicate that whether the sth edge estimator receives the sensory data of the ith sensor at the tth time step. For example, if the experienced delay is τ_i, then $\beta_i(t-h)|_{h=\tau_i} = 1$ and $\beta_i(t-h)|_{h\neq\tau_i} = 0$. Because of the experienced delay, the Kalman filter executed by the edge estimator is modified. In particular

$$\begin{aligned}\hat{x}_i(t) &= A_i(t-1)\hat{x}_i(t-1) + \beta_i(t)K_i(t)\\ &\quad\times\left[y_i(t) - C_i(t)A_i(t-1)\hat{x}_i(t-1)\right]\\ W_{ii}(t) &= [I_n - \beta_i(t)K_i(t)C_i(t)]W_{ii}^{-}(t).\end{aligned}$$

For notation simplicity, let $(\hat{x}_i(t), W_{ii}(t)) = \mathcal{KF}(\hat{x}_i(t-1), W_{ii}^{-}(t-1), \beta_i(t), y_i(t))$ represent the aforementioned state estimation. Due to the experienced delay, the sensory data obtained by edge estimators is delayed. In order to improve the estimation accuracy, the delayed sensory data is utilized to infer the corresponding edge estimate, i.e., calculating the edge estimate $\hat{x}_i(t-\tau_i)$ based on the previous sensory data $y_i(t-\tau_i)$. After that, the edge estimate $\hat{x}_i(t)$ is obtained by updating $\hat{x}_i(t-\tau_i)$. In detail

$$\begin{aligned}&\left(\hat{x}_i(t-\tau_i), W_{ii}(t-\tau_i)\right)\\ &= \mathcal{KF}\left(\hat{x}_i(t-\tau_i-1), W_{ii}^{-}(t-\tau_i-1), \beta_i(t-\tau_i), y_i(t-\tau_i)\right)\\ &\quad\ldots\quad\ldots\end{aligned} \tag{12}$$

$$\begin{aligned}&\left(\hat{x}_i(t-1), W_{ii}(t-1)\right)\\ &= \mathcal{KF}\left(\hat{x}_i(t-2), W_{ii}^{-}(t-2), \beta_i(t-1), y_i(t-1)\right)\end{aligned} \tag{13}$$

$$\begin{aligned}&\left(\hat{x}_i(t), W_{ii}(t)\right)\\ &= \mathcal{KF}\left(\hat{x}_i(t-1), W_{ii}^{-}(t-1), \beta_i(t), y_i(t)\right).\end{aligned} \tag{14}$$

For notation simplify, let $(\hat{x}_i(t), W_{ii}(t)) = \mathcal{KF}^{\tau_i}(\hat{x}_i(t-\tau_i-1), W_{ii}^{-}(t-\tau_i-1), \beta_i(t-\tau_i), y_i(t-\tau_i))$ represent the edge estimate $\hat{x}_i(t-\tau_i)$ that is calculated based on the sensory data $y_i(t-\tau_i)$. It can be seen that in case B, the edge estimation error exponentially increases with the growth of offloading delay, i.e., the impact of offloading delay directly reflects on the edge estimate.

The fusion estimate is $\hat{x}(t) = \sum_{i=1}^{N}\Omega_i'(t)\hat{x}_i(t|t-\tau_i)$, where the optimal weighted matrices $\{\Omega_1'(t), \ldots, \Omega_N'(t)\}$ are given by

$$\left[\Omega_1'(t), \ldots, \Omega_N'(t)\right] = \left[I_0^T W^{-1}(t) I_0\right]^{-1} I_0^T W^{-1}(t). \tag{15}$$

The cross error covariance matrix is $W_{ij}(t) = [I_n - \beta_i(t)K_i(t)C_i(t)][A_i(t-1)W_{ij}(t-1)A_i(t-1)^T + F(t-1)Q_w(t-1)F(t-1)^T][I_n - \beta_j(t)K_j(t)C_j(t)]^T$, then the error covariance matrix of fusion estimation is given by

$$\mathbb{E}\left\{\left(x(t)-\hat{x}(t)\right)\left(x(t)-\hat{x}(t)\right)^T\right\} = \left[I_0^T W^{-1}(t) I_0\right]^{-1}. \tag{16}$$

C. Joint Design of Task Offloading and Sensor Scheduling

In this work, the task offloading and sensor scheduling are jointly designed to minimize the mean square error for fusion estimation with delayed sensory data/local estimates. At each time step, the interested optimization problem is mathematically formulated as

$$\mathcal{P}_0: \min_{\{\delta(t),\theta(t),p(t)\}} \mathrm{Tr}\left(\mathbb{E}\left\{\left(x(t)-\hat{x}(t)\right)\left(x(t)-\hat{x}(t)\right)^T\right\}\right) \tag{17}$$

$$\text{s.t.}\quad \sum_{s=1}^{S}\delta_{i,s}(t) \le 1\ \forall i \tag{17a}$$

$$\sum_{i=1}^{N}\delta_{i,s}(t) \le 1\ \forall s \tag{17b}$$

$$p_i(t) \le p_{\max}\ \forall i \tag{17c}$$

$$\delta_{i,s}(t) \in \{0,1\}, \theta_i(t) \in \{0,1\}\ \forall i\ \forall s \tag{17d}$$

where $\hat{x}(t) = \hat{x}(t|t-\tau_i(t))$. Moreover, (17a) denotes one sensor can be scheduled by at most of one edge estimator; (17b) denotes the edge estimator can schedule at most of one sensor; and (17c) is the transmission power constraint of each sensor. In addition, $\mathrm{Tr}(\mathbb{E}\{(x(t)-\hat{x}(t))(x(t)-\hat{x}(t))^T\}) = \mathrm{Tr}(\Gamma_A(t)[I_0^T\Xi^{-1}(t)I_0]^{-1}) + \mathrm{Tr}(\Gamma_B(t)[I_0^T W^{-1}(t)I_0]^{-1})]$ with $\Gamma_A(t) = \mathrm{diag}\{\sum_{s\in\mathcal{S}}\delta_{i,s}(t)\ (1-\theta_i(t))\}$ and $\Gamma_B(t) = \mathrm{diag}\{\sum_{s\in\mathcal{S}}\delta_{i,s}(t)\ \theta_i(t)\}$. The optimization objective function includes the time-varying variable x with respect to the estimation step t. The system state changes step by step. At each estimation step, state x is time invariant. But, the state x changes based on the control command and the system dynamics from one estimation step to another estimation step. Taking the computational complexity and time-varying variables into account, the formulated problem aims to minimize the estimation error at each estimation step. For notation simplicity, in the remainder of this article, notation t is removed from the variables $\delta(t)$, $\theta(t)$, and $p(t)$, if there is no confusion.

IV. Estimation Task Offloading and Sensor Scheduling

For the formulated problem (17), the complicated objective function makes it intractable to solve $\mathcal{P}_0$. Thus, we will deal with the objective function $\mathcal{F}_0$ first in this section. According to the properties of matrix trace, we can obtain that

$$\begin{aligned}\mathcal{F}_0 &= \mathrm{Tr}\left(\Gamma_A(t)\left[I_0^T\Xi^{-1}(t)I_0\right]^{-1} + \Gamma_B(t)\left[I_0^T W^{-1}(t)I_0\right]^{-1}\right)\\ &= \mathrm{Tr}\left(\left[I_0^{-1}\Gamma_A(t)\Xi(t)\left(I_0^T\right)^{-1}\right] + \left[I_0^{-1}\Gamma_B(t)W(t)\left(I_0^T\right)^{-1}\right]\right)\\ &= \mathrm{Tr}\left(I_0^{-1}[\Gamma_A(t)\Xi(t) + \Gamma_B(t)W(t)]\left(I_0^T\right)^{-1}\right)\\ &\le \mathrm{Tr}([\Gamma_A(t)\Xi(t) + \Gamma_B(t)W(t)]) + \mathrm{Tr}\left(\left[\left(I_0^T\right)^{-1}I_0^{-1}\right]\right)\end{aligned}$$

where $\mathrm{Tr}([(I_0^T)^{-1}I_0^{-1}])$ is a known constant that is irrelevant to the scheduling variable δ, the offloading decision variable θ, and the power control variable p. Therefore, the objective function of the formulated problem (17) can be transformed into

$$\mathcal{F}_0 = \mathrm{Tr}(\Gamma_A(t)\Xi(t) + \Gamma_B(t)W(t)). \tag{18}$$

In addition, $\Gamma_A(t)$ and $\Gamma_B(t)$ have the same item, i.e., $\sum_{s=1}^{S}\delta_{i,s}(t)$, which denotes that whether the ith sensor is scheduled or not. If $\sum_{s=1}^{S}\delta_{i,s}(t) = 0$, no matter what the value of $\theta_i(t)$ is, $\Gamma_A(t)_{ii} = 0$ and $\Gamma_B(t)_{ii} = 0$. Thus, we first focus on how to determine the sensor scheduling without consideration of the task offloading.

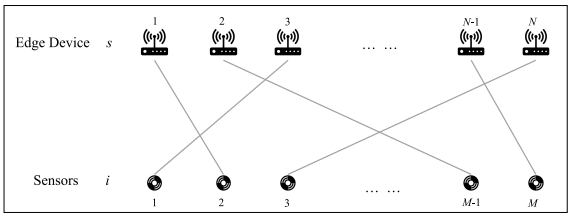

Fig. 4. Weighted bipartite graph for the matching problem of sensors and edge devices.

A. Sensor Scheduling

Let $\Lambda(t) = \text{diag}\{\sum_{s\in\mathcal{S}} \delta_{i,s}(t)\}$, $\tilde{\Xi}(t) = \text{diag}\{(1-\theta_i(t))\}\Xi(t)$, $\tilde{W}(t) = \text{diag}\{\theta_i(t)\}W(t)$, and $\tilde{M}(t) = \tilde{\Xi}(t) + \tilde{W}(t)$. Then, (18) is rewritten as $\text{Tr}(\Lambda(t)\tilde{\Xi}(t) + \Lambda(t)\tilde{W}(t))$ by assuming that $\theta(t)$ and $p(t)$ is given. As aforementioned, if the ith sensor is not scheduled ($\sum_{s=1}^{S} \delta_{i,s}(t) = 0$), the experienced delay is set to be $\mathcal{T}$. Therefore, if $\Lambda_{ii}(t) = 0$, $\tilde{M}_{ii}(t) = \tilde{\Xi}_{ii}(t-\mathcal{T}) + \tilde{W}_{ii}(t-\mathcal{T})$. Otherwise, $\tilde{M}_{ii}(t) = \tilde{\Xi}_{ii}(t) + \tilde{W}_{ii}(t)$. Since $\tilde{\Xi}_{ii}(t-\tau_i(t)) > \tilde{\Xi}_{ii}(t)$ and $\tilde{W}_{ii}(t-\tau_i(t)) > \tilde{W}_{ii}(t)$, the objective function of $\mathcal{P}_1$ reaches the minimum value when the S minimum $\tilde{M}_{ii}(t)$'s are selected. In this regard, the subproblem of sensor scheduling is given by

$$\mathcal{SP}_1: \min_{\{\delta(t)\}} \sum_{i=1}^{N}\sum_{s=1}^{S} \delta_{i,s}(t)\tau_{i,s}(t) \tag{19}$$

$$\text{s.t.} \quad \sum_{s=1}^{S} \delta_{i,s}(t) \le 1 \ \forall i \tag{19a}$$

$$\sum_{i=1}^{N} \delta_{i,s}(t) = 1 \ \forall s \tag{19b}$$

$$\delta_{i,s}(t) \in \{0,1\} \ \forall i \ \forall s. \tag{19c}$$

The binary integer programming problem could be optimally solved with the algorithms based on exhaustive search at the cost of exponential complexity. In this work, we find out that the binary linear programming problem $\mathcal{P}_1$ could be regarded as an optimal matching problem of weighted bipartite graph, which is conducive to solving it efficiently. As shown in Fig. 4, the set of sensors and the set of edge estimators are two mutually disjointed vertex sets in the bipartite graph. If the sth edge estimator schedules the ith sensor, there is a line between the sth vertex and the ith vertex. The Kuhn–Munkres (KM) [29] algorithm is regarded as an effective binary matching algorithm. The KM algorithm is a minimization algorithm proposed by Kuhn [30] and then enhanced by Munkres [31].

As the KM algorithm is an effective method to solve the weighted matching [32], in this work, the KM algorithm is employed to deal with the sensor scheduling problem. Based on the KM theorem, the optimization problem of finding a min-weight matching is transformed into a combinatorial optimization problem of finding a perfect matching [33]. In general, a weighted bipartite graph is defined as $\mathbb{G}(\mathcal{V}, \mathcal{E}, \mathcal{W})$. The set of vertices is denoted as $\mathcal{V} = \mathcal{N} \cup \mathcal{M}$, where $\mathcal{N}$ is the set of sensors and $\mathcal{M}$ is the set of edge devices. Moreover, $\mathcal{E}$ is the set of edges connecting a vertex in set $\mathcal{N}$ and a vertex in set $\mathcal{M}$. The weight of edges is denoted as the vector $\mathcal{W}$. In this work, the weight of the line connecting the sth vertex and the ith vertex is set as $-\sum_{s=1}^{S} \delta_{i,s}(t)\tau_{i,s}(t)$. In this way, the optimal solution of the original minimum weight matching problem is obtained by finding the optimal matching in the bipartite with the KM algorithm. The KM algorithm could obtain the optimal sensor scheduling with polynomial computational complexity $\mathcal{O}(N)^3$.

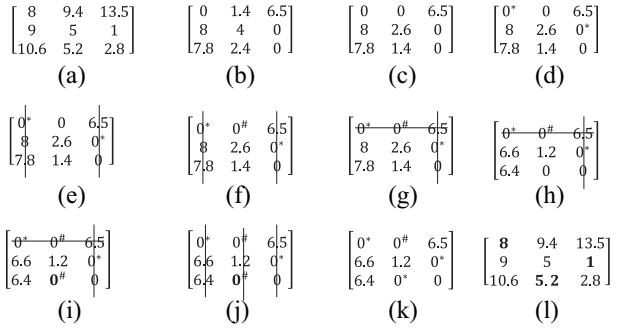

Fig. 5. Example of the KM algorithm. (a) Original matrix. (b) 1st step. (c) 2nd step. (d) 3rd step. (e) 4th step. (f) 5th step. (g) 6th step. (h) 7th step. (i) 8th step. (j) 9th step. (k) 10th step. (l) 11th step.

The example is shown as Fig. 5. In particular, the matrix is set to be $Y = [8, 9.4, 13.5; 9, 5, 1; 10.6, 5.2, 2.8]$, shown in Fig. 5(a). The main idea of the KM algorithm is to find out the smallest element in each row at first. Then, the smallest element is subtracted from all elements in this row, and the same process is done for all columns. After that, all zeros are separated into two sets. One set is consisted of "starred zeros," and the other is constructed with possible "candidate zeros." In particular, the smallest element in each row is "8," "1," and "2.8," then all elements in this row subtract the smallest one, shown in Fig. 5(b). Based on this result, the smallest element in each column is "0," "1.4," and "0," respectively. Similarly, all elements in this column subtract the smallest one, shown in Fig. 5(c). Then, mark the zero element with star for all rows or column that has no starred zeros, shown in Fig. 5(d). It can be seen that, the starred zeros only cover column 1 and column 2, shown in Fig. 5(e). Therefore, let the element, that is zero but is not set to be starred zero, be the candidate zero, shown in Fig. 5(f). After that, the row with starred zero and candidate zero is considered to be covered, and the column with starred zero is also covered, shown in Fig. 5(g). In this case, the smallest one among rest elements is "1.4," the 1.4 is subtracted from all rest elements, shown in Fig. 5(h). As in Fig. 5(i), the zero elements is highlighted. Then we need to check that whether all columns are covered with highlighted candidate zero and starred zeros. The result is shown in Fig. 5(k). Thus, the highlighted candidate zero is changed to be starred zero, shown in Fig. 5(l). Finally, in Fig. 5(m), the obtained allocation result is [0, 0], [1, 2], [2, 1], and the minimum sum $8 + 1 + 5.2 = 14.2$.

B. Task Offloading With Power Control

In this section, the aim of task offloading is to minimize $\tilde{M}_{ii}(t) = (1-\theta_i(t))\Xi_{ii}(t) + \theta_i(t)W_{ii}(t)$ for scheduled sensors, where $\Xi_{ii}(t)$ is the local estimation error at the ith sensor and $W_{ii}(t)$ is the edge estimation error for the ith sensor. If $||e_i(t-(\tau_i^l + \tau_{i,s}^{lt}(t))||_2^2 \ge W_{ii}(t-(\tau_{i,s}^e + \tau_{i,s}^{et}(t)))$, the value of $\theta_i(t)$ prefers to be one, which means that the sensor i expects to be offload the estimation task to an edge estimator. Otherwise, the

sensor i will execute the state estimation locally. Therefore, the decision of task offloading relies on the impact of experienced delay on the estimation error. In addition, the experienced delay depends on packet length of delivered information, computation ability, and achievable data rate. When the sensor scheduling is given, the values of τ_i^l and $\tau_{i,s}^e$ are known. Therefore, the achievable data rate is the only adjustable value that effects the offloading decision. In addition, (5) shows that the transmission power effects the achievable data rate, thus the power control is performed to reduce the experienced delay. Then, the power control problem of the ith sensor is formulated as

$$\begin{aligned}\mathcal{SP}_2:\ \min_{\{p\}}\ & \left[\left(1-\sum_{s=1}^{S}\delta_{i,s}(t)\right)\mathcal{T}+\sum_{s=1}^{S}\delta_{i,s}(t)\frac{l_i}{r_{i,s}(t)}\right]\\ \text{s.t.}\ & p_i(t)\le p_{\max}\ \forall i\\ & r_{i,s}(t)=\log\left(1+\frac{p_i(t)g_{i,s}(t)}{\sum_{i'\neq i,i'\in\mathcal{N}}p_{i'}(t)g_{i',s}(t)+N_0}\right).\end{aligned}\tag{20}$$

For ease of handling, this constrained nonlinear problem is rewritten as

$$\begin{aligned}\min_{\{p\}}\ & \mathcal{D}\\ \text{s.t.}\ & p_i(t)\le p_{\max}\ \forall i\\ & \left(1-\sum_{s=1}^{S}\delta_{i,s}(t)\right)\mathcal{T}+\sum_{s=1}^{S}\delta_{i,s}(t)\frac{l_i}{r_{i,s}(t)}\le\mathcal{D}\ \forall i\end{aligned}\tag{21}$$

where $r_{i,s}(t) = \log(1 + ([p_i(t)g_{i,s}(t)]/[\sum_{i'\neq i,i'\in\mathcal{N}} p_{i'}(t)g_{i',s}(t) + N_0]))$.

We then derive an equivalent reformulation of (21) to expose the hidden convexity. With the given $\delta_{i,s}(t)$, the power control is executed only for the scheduled sensors, i.e., $\sum_{s=1}^{S}\delta_{i,s}(t) = 1$. Moreover, the edge estimator is denoted as $\hat{s} = \{s|\delta_{i,s}(t) = 1\ \forall\ s\}$, since a sensor could be scheduled by at most of one edge estimator. Then, (21) is rewritten as

$$\min_{\{p,\mathcal{D},\mathcal{R},\mathcal{H},\mathcal{I}\}}\ \mathcal{D}\tag{22}$$

$$\begin{aligned}\text{s.t.}\ & p_i(t)\le p_{\max}\ \forall i && \text{(22a)}\\ & l_i/\mathcal{R}_i\le\mathcal{D}\ \forall i && \text{(22b)}\\ & \log(1+\mathcal{H}_i)\ge\mathcal{R}_i\ \forall i && \text{(22c)}\\ & p_i(t)g_{i,s}(t)/\mathcal{I}_i\ge\mathcal{H}_i\ \forall i && \text{(22d)}\\ & \sum_{i'\neq i,i'\in\mathcal{N}}p_{i'}(t)g_{i',s}(t)+N_0\le\mathcal{I}_i\ \forall i && \text{(22e)}\end{aligned}$$

which obtain the optimal solution when all constraints hold with equality [34]. In this case, problem (21) is equivalent to (22). In order to solve (22), we first deal with the nonconvex constraints (22d). It can be seen that the left-hand side of (22d) is convex and the right-hand sides is an affine function, thus (22d) is jointly convex with respect to the involved variables. Moreover, it is appropriate to apply the inner approximation algorithm [35] to deal with the nonconvex part of (22d). In particular, we approximate (22d) with a linear constraint $\mathcal{J}_i^{(n)}(p_i(t),\mathcal{I}_i)\ge\mathcal{H}_i$, which is the first order of $p_i(t)g_{i,s}(t)/\mathcal{I}_i$ around $p_i^{(n)}(t)$ and $\mathcal{I}_i^{(n)}$, i.e.,

$$\mathcal{J}_i^{(n)}(p_i(t),\mathcal{I}_i)=\frac{2p_i^{(n)}(t)g_{i,s}(t)g_{i,s}(t)}{\mathcal{I}_i^{(n)}}p_i(t)-\frac{\left[p_i^{(n)}(t)g_{i,s}(t)\right]^2}{\left[\mathcal{I}_i^{(n)}\right]^2}\mathcal{I}_i\tag{23}$$

where the superscript n is the nth iteration of the iterative algorithm. For the iteration $n+1$, the approximate problem is convex, which is given by

$$\begin{aligned}\min_{\{p,\mathcal{D},\mathcal{R},\mathcal{H},\mathcal{I}\}}\ & \mathcal{D}\\ \text{s.t.}\ & \mathcal{J}_i^{(n)}(p_i(t),\mathcal{I}_i)\ge\mathcal{H}_i\ \forall i\\ & \text{(22a)}-\text{(22c)},\ \text{(22e)}.\end{aligned}\tag{24}$$

In detail, we first solve problem (24) with $(p^{(n-1)},\mathcal{I}^{(n-1)})$, then the optimal solution $(p^*,\mathcal{I}^*)$ could be obtained. Finally,update $(p^{(n)},\mathcal{I}^{(n)})$ with $(p^*,\mathcal{I}^*)$. The iteration process is convergent. The main reason is that the optimal solution obtained at the nth iteration is a feasible solution of the problem at the $n+1$ iteration. Therefore, the sequence of obtained solutions in the iteration process is nonincreasing $(\mathcal{D}^{(n-1)}\le\mathcal{D}^{(n)})$, and the iteration process is convergent.

Based on the value of $r_{i,s}^o(t)$ that is obtained by solving (21), the ith sensor can determine the relative relationship between $||e_i(t-(\tau_i^l+(l_i/r_{i,s}^o(t))))||_2^2$ and $W_{ii}(t-(\tau_{i,s}^e+(l_i^e/r_{i,s}^o(t))))$. If $||e_i(t-(\tau_i^l+(l_i/r_{i,s}^o(t))))||_2^2 > W_{ii}(t-(\tau_{i,s}^e+(l_i^e/r_{i,s}^o(t))))$, $\theta_i(t)=1$. Otherwise, $\theta_i(t)=0$.

In summary, the proposed original problem is a mixed-integer nonlinear programming, whose optimal solution is challenging to obtain. The decomposition method is used to obtain a suboptimal solution of the original problem effectively. The original problem is separated into two subproblems. One is the sensor scheduling subproblem, and the other is the task offloading with power control problem. The former one is a binary integer programming problem, which is optimally solved with the designed polynomial-time algorithm. As the latter one includes a nonconvex constraint, the inner approximation algorithm is used to deal with the nonconvex part. Moreover, the latter one is suboptimally solved in an iterative manner. The detailed solution process is shown in Fig. 6.

V. Performance Evaluation

In this article, the proposed AES approach jointly designs the estimation task offloading and the sensor scheduling for edge-computing enabled state estimation to achieve effective state estimation with limited network resources. In this section, we conduct extensive simulations to evaluate the performance of AES approach in terms of mean square error, experienced delay, response rate, where the response rate is defined as the ratio of the number of sensors within the tolerant delay $\mathcal{T}$ to the total number of requests [36]. For comparison purpose, two other schemes are implemented in the following simulations. One is the distributed sensing (DS) approach, wherein each sensor performs the local estimation and deliver the local estimate to edge estimators. The other is the centralized sensing (CS) approach, wherein each sensor directly deliver the

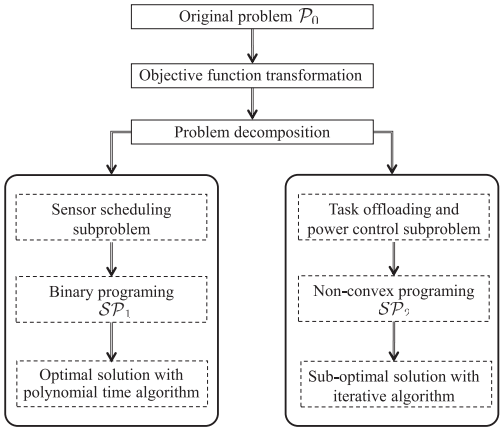

Fig. 6. Diagram of the solution process.

TABLE I
MAIN PARAMETERS

Parameters	Values
Reference distance d_0	1 m
Path loss at reference distance PL_0	56.7 dB
Noise power N_0	-87 dBm
Path loss exponent n	2.31
Small-scale fading distribution between sensor and edge device	Rayleigh (0,1)
Computing capability of sensor	Randomly from 0.7-1.5 GHz
Computing capability of edge device	Randomly from 1-3 GHz
Packet length of the raw measurement	Randomly from 300-500 bits
Packet length of the local estimate	Randomly from 100-300 bits
Maximum transmit power of sensor	50 mw
Bandwidth of each channel	0.2 MHz
Sampling interval of the estimator $\mathcal{T}$	100 ms

raw measurements to edge estimators without any processing. The performance is evaluated according to the network topology covering a rectangular area $[0, 100]_m \times [0, 100]_m$, wherein sensors and edge estimators are randomly placed. The main parameters used in simulations, are shown in Table I. Set the initial state covariance matrix and the initial error covariance matrix as $P_0 = 10\mathbf{I}$, $W_0 = 10\mathbf{I}$, and $Q = 10\mathbf{I}$, respectively, where $\mathbf{I}$ is a unit diagonal matrix [37].

A. Performance Evaluation Among Compared Approaches

The performance comparison in terms of MSE among three compared approaches is shown in Fig. 7. It can be seen that the mean value of state estimation error with the proposed AES approach is less than that of comparing ones. This is because that in the proposed AES approach, each sensor makes the task offloading decision based on both the wireless channel conditions and the computational capacity to minimize the state estimation error. Therefore, the proposed AES approach achieved a communication–computation tradeoff, making it possible to perform a more accurate state estimation.

Considering the random properties of wireless channels, the statistic characteristic analysis of three compared approaches is shown in Fig. 8. Fig. 8(a) shows that for AES approach, most of the samples fall in the interval [5, 10]. However, for DS and CS approaches, most of the samples fall in the interval [6, 11] and [5, 14], respectively. Moreover, it can be seen that the blue line in Fig. 8(b) is on the left, thus the state estimation error

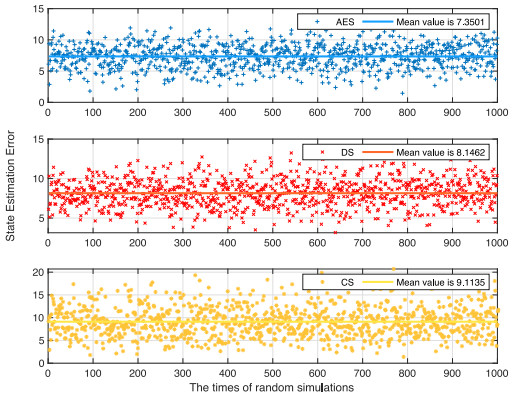

Fig. 7. State estimation error comparison among different approaches.

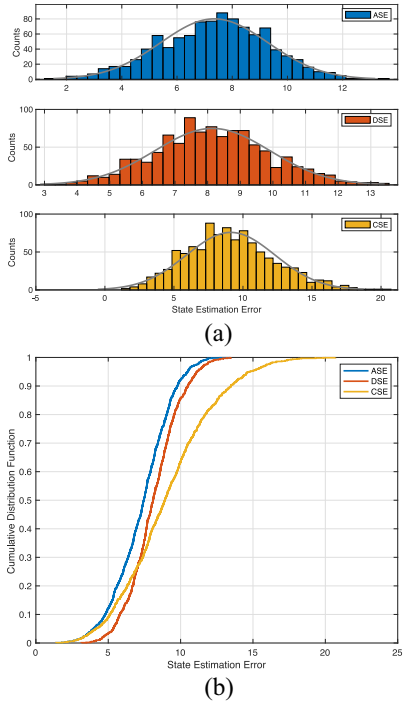

Fig. 8. Statistic characteristic analysis. (a) Statistical counting. (b) Cumulative distribution function.

of AES approach is less than that of DS and CS approaches for any cumulative distribution probability.

B. Performance Comparison With Different Parameters

We introduce two variables to denote the impact of computational capacity and packet length on the perform evaluation, respectively. The computational capacity ratio is defined as the ration between the average computational capacity of edge estimators and that of sensors. Moreover, with the assistance of edge computing, the data volume of exchanged sensing information could be significantly reduced. The packet length ratio is used to quantitatively evaluate the reduction of data volume. The packet length ratio is defined as the average packet length of sensory data and that of local estimates. The larger the packet length ratio is, the more the data volume is reduced. In simulations, the length ratio is adjusted by changing the length of local estimate by fixing the length of sensory data.

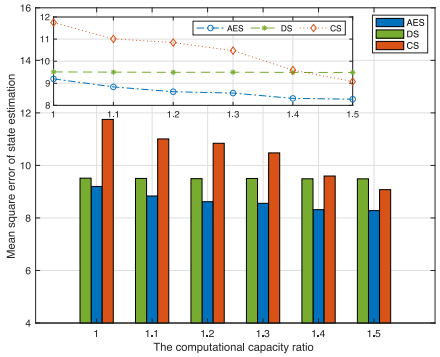

Fig. 9. Impact of computational capacity ratio on estimation error.

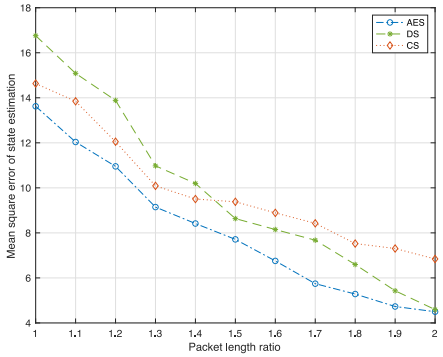

Fig. 11. Impact of packet length ratio on estimation error.

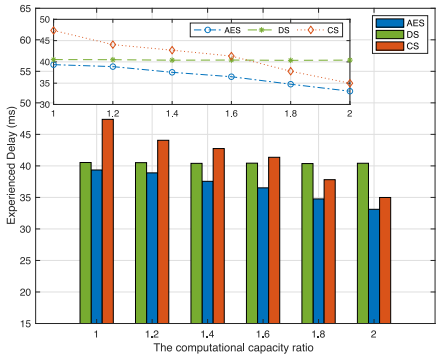

Fig. 10. Impact of computational capacity ratio on experienced delay.

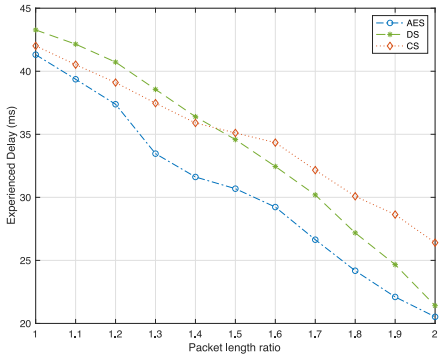

Fig. 12. Impact of packet length ratio on experienced delay.

The impact of computational capacity on the state estimation error are shown in Figs. 9 and 10, where the computational ratio is adjusted by changing the computational capacity of edge estimators. It can be seen that the blue curve is always below than two compared ones, which means that the proposed AES approach has advantages on reducing the estimation error and the experienced delay. Moreover, the state estimation errors achieved by the AES approach and CS approach decrease with the growth of the computational capacity ratio. However, the state estimation error achieved by DS approach does not change with the computational capacity ratio. In the DS approach, the state estimation is performed at sensors locally, thus the computational capacity of edge estimators almost have no effect on the estimation error or the experienced delay. Furthermore, we find out that there is a critical point of computational capacity ratio. If the computational capacity ratio is less than this critical point, the DS approach is better than the CS approach in terms of both the estimation error and the experienced delay. The reason is that with the growth of the computational capacity of edge estimators, the CS approach could archive the state estimation task at the lower computational delay. Therefore, when the edge estimators have high computational capacity, the computational delay of CS approach is small enough to counteract the impact of the large transmission delay of sensory on estimation accuracy.

In order to evaluate the impact of packet length ratio on the estimation error and experienced delay, the length ratio is adjusted by changing the length of local estimate but fixing the length of sensory data. The simulation results are shown in Figs. 11 and 12. It can be seen that both the estimation error and the experienced delay reduce with the increasing of the packet length ratio for all compared approaches. This is because that the shorter packets will cost less transmission delay and computational delay. It is worth noting that the proposed AES approach could archived the smallest estimation error and experienced delay, since it makes the task offloading decision based on the channel condition and computational capacity to make the communication–computing tradeoff for improving the state estimation accuracy with limited resources. Moreover, the decreasing rate of DS approach is larger than that of the CSE approach, since the shorter packet length of local estimate will significantly reduce the transmission delay in the DS approach. We also find out that there exists a turning point. When the packet length ratio is less than the turning point, the CS approach is better than the DS approach. Otherwise, the DS approach is better than the CS approach. This phenomenon is caused by that when the packet length of local estimate is smaller, the advantage of edge estimation on reducing the transmission delay is more significant.

Earlier in this section, we have discussed the estimation error and offloading delay with different computational capacity ratios and packet length ratios. Then, we will discuss the relationship between offloading delay and estimation error, which is shown in Fig. 13. It can be seen that the estimation error exponentially increase with the growth of offloading delay, meaning that the offloading delay reduction could decrease the estimation error. Moreover, we find that the estimation error is delay sensitive when offloading delay is large. In other words, for the large offloading delay, a little growth of offloading delay may lead to a significantly large increase of estimation error, thus it is necessary to design the task offloading and sensor scheduling schemes

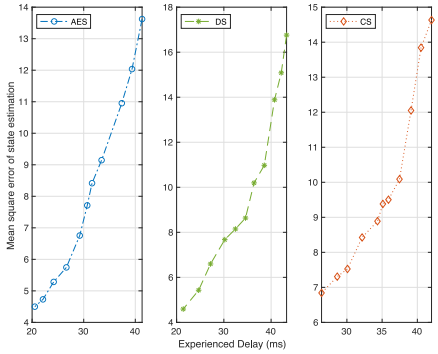

Fig. 13. Relationship between experienced delay and estimation error.

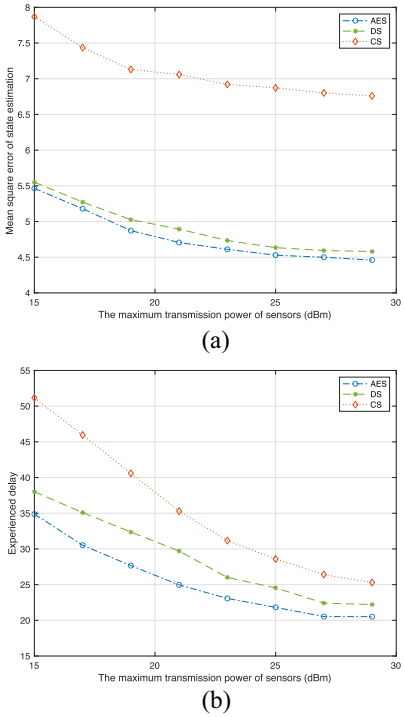

Fig. 14. Impact of maximum transmission power. (a) Estimation error. (b) Experienced delay.

to minimize the estimation error rather than the offloading delay.

At the end of this section, we discuss the impact of maximum transmission power on the estimation error and the offloading delay, shown in Fig. 14. It can be seen that both the estimation error and the offloading delay decrease with the growth of maximum transmission power at first, and then hold steady. It reveals that only increasing the maximum transmission power could not continuously improve the estimation accuracy, since the estimation performance not only depends on the transmission power but also depends on the sensitivity to offloading delay.

VI. Conclusion

We have investigated the joint design and optimization of estimation task offloading and sensor scheduling to achieve an AES for industrial IoT systems. In particular, the relationship between estimation error and offloading delay has been derived to indicate the necessary of adjusting offload delay on demand for reducing estimation error. Then, an AES scheme has been proposed to minimize the estimation error by jointly optimizing task offloading and sensor scheduling. Finally, simulation results have been provided to demonstrate that the proposed scheme has realized a smaller estimation error than compared ones. The proposed scheme could adaptively adjust the estimation task offloading decision based on the communication and computation capacity of sensors, which is appropriate for performance improvement in edge-enabled industrial IoT systems. For future work, we will study the distributed consensus estimation for industrial IoT systems over lossy wireless channels.

References

[1] J. Gao, W. Zhuang, M. Li, X. Shen, and X. Li, "MAC for machine-type communications in industrial IoT—Part I: Protocol design and analysis," *IEEE Internet Things J.*, vol. 8, no. 12, pp. 9945–9957, Jun. 2021.

[2] N. Cheng *et al.*, "Space/aerial-assisted computing offloading for IoT applications: A learning-based approach," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 5, pp. 1117–1129, May 2019.

[3] Z. Yang, R. Wang, D. Wu, H. Wang, H. Song, and X. Ma, "Local trajectory privacy protection in 5G enabled industrial intelligent logistics," *IEEE Trans. Ind. Informat.*, vol. 18, no. 4, pp. 2868–2876, Apr. 2022.

[4] L. Lyu, C. Chen, S. Zhu, N. Cheng, B. Yang, and X. Guan, "Control performance aware cooperative transmission in multiloop wireless control systems for industrial IoT applications," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3954–3966, Oct. 2018.

[5] S. R. Khosravirad, H. Viswanathan, and W. Yub, "Exploiting diversity for ultra-reliable and low-latency wireless control," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 316–331, Jan. 2021.

[6] J. Yang *et al.*, "Ultra-reliable communications for Industrial Internet of Things: Design considerations and channel modeling," *IEEE Netw.*, vol. 33, no. 4, pp. 104–111, Jul./Aug. 2019.

[7] F. Tramarin, A. K. Mok, and S. Han, "Real-time and reliable industrial control over wireless LANs: Algorithms, protocols, and future directions," *Proc. IEEE*, vol. 107, no. 6, pp. 1027–1052, Jun. 2019.

[8] C. She, C. Liu, T. Q. Quek, C. Yang, and Y. Li, "Ultra-reliable and low-latency communications in unmanned aerial vehicle communication systems," *IEEE Trans. Commun.*, vol. 67, no. 5, pp. 3768–3781, May 2019.

[9] C. M. Nguyen, P. N. Pathirana, and H. Trinh, "Robust state estimation for non-linear systems with unknown delays," *IET Control Theory Appl.*, vol. 13, no. 8, pp. 1147–1154, 2019.

[10] J. Wang, S. Ma, C. Zhang, and M. Fu, "H_∞ state estimation via asynchronous filtering for descriptor Markov jump systems with packet losses," *Signal Process.*, vol. 154, pp. 159–167, Jan. 2019.

[11] X.-M. Li, B. Zhang, P. Li, Q. Zhou, and R. Lu, "Finite-horizon H_∞ state estimation for periodic neural networks over fading channels," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 5, pp. 1450–1460, May 2020.

[12] Y. Xu, R. Lu, P. Shi, J. Tao, and S. Xie, "Robust estimation for neural networks with randomly occurring distributed delays and Markovian jump coupling," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 4, pp. 845–855, Apr. 2018.

[13] L. Lyu *et al.*, "Dynamics-aware and beamforming-assisted transmission for wireless control scheduling," *IEEE Trans. Wireless Commun.*, vol. 17, no. 11, pp. 7677–7690, Nov. 2018.

[14] M. Rana, L. Li, and S. W. Su, "Distributed state estimation over unreliable communication networks with an application to smart grids," *IEEE Trans. Green Commun. Netw.*, vol. 1, no. 1, pp. 89–96, Mar. 2017.

[15] W.-A. Zhang, G. Feng, and L. Yu, "Multi-rate distributed fusion estimation for sensor networks with packet losses," *Automatica*, vol. 48, no. 9, pp. 2016–2028, 2012.

[16] Z. Li, N. Zhu, D. Wu, H. Wang, and R. Wang, "Energy-efficient mobile edge computing under delay constraints," *IEEE Trans. Green Commun. Netw.*, vol. 6, no. 2, pp. 776–786, Jun. 2022.

[17] L. Lin, X. Liao, H. Jin, and P. Li, "Computation offloading toward edge computing," *Proc. IEEE*, vol. 107, no. 8, pp. 1584–1607, Aug. 2019.

[18] M. Sheng, Y. Dai, J. Liu, N. Cheng, X. Shen, and Q. Yang, "Delay-aware computation offloading in NOMA MEC under differentiated uploading delay," *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2813–2826, Apr. 2020.

[19] H. He, H. Shan, A. Huang, Q. Ye, and W. Zhuang, "Edge-aided computing and transmission scheduling for LTE-U-enabled IoT," *IEEE Trans. Wireless Commun.*, vol. 19, no. 12, pp. 7881–7896, Dec. 2020.

[20] Y. Pan, M. Chen, Z. Yang, N. Huang, and M. Shikh-Bahaei, "Energy-efficient NOMA-based mobile edge computing offloading," *IEEE Commun. Lett.*, vol. 23, no. 2, pp. 310–313, Feb. 2019.
[21] Y. Dai, D. Xu, S. Maharjan, and Y. Zhang, "Joint computation offloading and user association in multi-task mobile edge computing," *IEEE Trans. Veh. Technol.*, vol. 67, no. 12, pp. 12313–12325, Dec. 2018.
[22] Q. Ye, W. Shi, K. Qu, H. He, W. Zhuang, and X. Shen, "Joint RAN slicing and computation offloading for autonomous vehicular networks: A learning-assisted hierarchical approach," *IEEE Open J. Veh. Technol.*, vol. 2, pp. 272–288, Jun. 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9454395
[23] J. Zhang *et al.*, "Energy-latency tradeoff for energy-aware offloading in mobile edge computing networks," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2633–2645, Aug. 2018.
[24] C. Wang, C. Liang, F. R. Yu, Q. Chen, and L. Tang, "Computation offloading and resource allocation in wireless cellular networks with mobile edge computing," *IEEE Trans. Wireless Commun.*, vol. 16, no. 8, pp. 4924–4938, Aug. 2017.
[25] Z. Kuang, L. Li, J. Gao, L. Zhao, and A. Liu, "Partial offloading scheduling and power allocation for mobile edge computing systems," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6774–6785, Aug. 2019.
[26] M. Nourian, A. S. Leong, and S. Dey, "Optimal energy allocation for Kalman filtering over packet dropping links with imperfect acknowledgments and energy harvesting constraints," *IEEE Trans. Autom. Control*, vol. 59, no. 8, pp. 2128–2143, Aug. 2014.
[27] L. Lyu, C. Chen, S. Zhu, and X. Guan, "5G enabled codesign of energy-efficient transmission and estimation for industrial IoT systems," *IEEE Trans. Ind. Informat.*, vol. 14, no. 6, pp. 2690–2704, Jun. 2018.
[28] B. Chen, W.-A. Zhang, and L. Yu, "Distributed finite-horizon fusion Kalman filtering for bandwidth and energy constrained wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 62, no. 4, pp. 797–812, Feb. 2014.
[29] H. Zhu, M. Zhou, and R. Alkins, "Group role assignment via a Kuhn–Munkres algorithm-based solution," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 42, no. 3, pp. 739–750, May 2012.
[30] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Res. Logist. Quart.*, vol. 2, nos. 1–2, pp. 83–97, 1955.
[31] J. Munkres, "Algorithms for the assignment and transportation problems," *J. Soc. Ind. Appl. Math.*, vol. 5, no. 1, pp. 32–38, 1957.
[32] D. Feng, L. Lu, Y. Yuan-Wu, G. Y. Li, G. Feng, and S. Li, "Device-to-device communications underlaying cellular networks," *IEEE Trans. Commun.*, vol. 61, no. 8, pp. 3541–3551, Aug. 2013.
[33] S. Mu, Z. Zhong, D. Zhao, and M. Ni, "Joint job partitioning and collaborative computation offloading for Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 1046–1059, Feb. 2019.
[34] K.-G. Nguyen, Q.-D. Vu, M. Juntti, and L.-N. Tran, "Distributed solutions for energy efficiency fairness in multicell MISO downlink," *IEEE Trans. Wireless Commun.*, vol. 16, no. 9, pp. 6232–6247, Sep. 2017.
[35] B. R. Marks and G. P. Wright, "A general inner approximation algorithm for nonconvex mathematical programs," *Oper. Res.*, vol. 26, no. 4, pp. 681–683, 1978.
[36] S. Hu and G. Li, "Dynamic request scheduling optimization in mobile edge computing for IoT applications," *IEEE Internet Things J.*, vol. 7, no. 2, pp. 1426–1437, Feb. 2020.
[37] L. Lyu, C. Chen, C. Hua, S. Zhu, and X. Guan, "Co-design of stabilisation and transmission scheduling for wireless control systems," *IET Control Theory Appl.*, vol. 11, no. 11, pp. 1767–1778, 2017.

Ling Lyu (Member, IEEE) received the B.S. degree in telecommunication engineering from Jinlin University, Changchun, China, in 2013, and the Ph.D. degree in control theory and control engineering from Shanghai Jiao Tong University, Shanghai, China, in 2019.

She joined Dalian Maritime University, Dalian, China, in 2019, where she is currently an Associate Professor with the School of Information Science and Technology. She is also with the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an, China. She was a visiting student with the University of Waterloo, Waterloo, ON, Canada, from September 2017 to September 2018. Her current research interests include wireless sensor and actuator network and application in industrial automation, the joint design of communication and control in industrial cyber–physical systems, estimation and control over lossy wireless networks, machine-type communication-enabled reliable transmission in the fifth generation network, resource allocation, and energy efficiency.

Lihong Zhao received the B.Eng. degree in electronic information engineering from Shandong University of Technology, Zibo, China, in 2020. He is currently pursuing the M.Eng. degree with Dalian Maritime University, Dalian, China.

His research interests include resource allocation and state estimation.

Yanpeng Dai (Member, IEEE) received the B.Eng. degree from Shandong Normal University, Zibo, China, in 2014, and the Ph.D. degree from Xidian University, Xi'an, China, in 2020.

He is currently an Assistant Professor with the College of Information Science and Technology, Dalian Maritime University, Dalian, China. His research interests include resource management and interference coordination of heterogeneous wireless networks.

Nan Cheng (Member, IEEE) received the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada, in 2016, and the B.E. and M.S. degrees from the Department of Electronics and Information Engineering, Tongji University, Shanghai, China, in 2009 and 2012, respectively.

He worked as a Postdoctoral Fellow with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada, from 2017 to 2019. He is currently a Professor with the State Key Laboratory of Integrated Services Networks and the School of Telecommunications Engineering, Xidian University, Xi'an, Shaanxi, China. His current research focuses on B5G/6G, space–air–ground-integrated network, big data in vehicular networks, and self-driving system. His research interests also include performance analysis, MAC, opportunistic communication, and application of AI for vehicular networks.

Cailian Chen (Member, IEEE) received the B.Eng. and M.Eng. degrees in automatic control from Yanshan University, Qinhuangdao, China, in 2000 and 2002, respectively, and the Ph.D. degree in control and systems from the City University of Hong Kong, Hong Kong SAR, in 2006.

She has been with the Department of Automation, Shanghai Jiao Tong University, Shanghai, China, since 2008. She is currently a Distinguished Professor. She has authored three research monographs and over 100 referred international journal papers. She is the inventor of more than 30 patents. Her research interests include industrial wireless networks and computational intelligence, and Internet of Vehicles.

Dr. Chen received the prestigious IEEE TRANSACTIONS ON FUZZY SYSTEMS Outstanding Paper Award in 2008, and five conference best paper awards. She won the Second Prize of National Natural Science Award from the State Council of China in 2018, the First Prize of Natural Science Award from The Ministry of Education of China in 2006 and 2016, respectively, and the First Prize of Technological Invention of Shanghai Municipal, China, in 2017. She was honored the National Outstanding Young Researcher by NSF of China in 2020 and the Changjiang Young Scholar in 2015. She has been actively involved in various professional services. She is a Distinguished Lecturer of IEEE VTS. She serves as a Deputy Editor for the *National Science Open*, and an Associate Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, *IET Cyber-Physical Systems: Theory and Applications*, and *Peer-to-Peer Networking and Applications* (Springer). She also served as a TPC Chair for ISAS19, a Symposium TPC Co-Chair for IEEE Globecom 2016, and a Track Co-Chair for VTC2016-fall and VTC2020-fall.

Xinping Guan (Fellow, IEEE) received the B.Sc. degree in mathematics from Harbin Normal University, Harbin, China, in 1986, and the Ph.D. degree in control science and engineering from Harbin Institute of Technology, Harbin, in 1999.

He is currently a Chair Professor with Shanghai Jiao Tong University, Shanghai, China, where he is the Dean of the School of Electronic, Information and Electrical Engineering, and the Director of the Key Laboratory of Systems Control and Information Processing, Ministry of Education of China. Before that, he was the Executive Director of the Office of Research Management, Shanghai Jiao Tong University, and a Full Professor and a Dean of Electrical Engineering with Yanshan University, Qinhuangdao, China. He has authored and/or coauthored five research monographs, more than 200 papers in IEEE TRANSACTIONS and other peer-reviewed journals, and numerous conference papers. As a Principal Investigator, he has finished/been working on more than 20 national key projects. He is the Leader of the prestigious Innovative Research Team of the National Natural Science Foundation of China. His current research interests cover industrial network systems, smart manufacturing, and underwater networks.

Dr. Guan received the Second Prize of the National Natural Science Award of China in both 2008 and 2018, the First Prize of Natural Science Award from the Ministry of Education of China in 2006 and 2016. He was a recipient of the IEEE TRANSACTIONS ON FUZZY SYSTEMS Outstanding Paper Award in 2008 and the IEEE TCCPS Industrial Technical Excellence Award in 2022. He is an Executive Committee Member of the Chinese Automation Association Council and the Chinese Artificial Intelligence Association Council. He is a National Outstanding Youth honored by NSF of China, the Changjiang Scholar by the Ministry of Education of China, and the State-Level Scholar of the New Century Bai Qianwan Talent Program of China.

Xuemin Shen (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990.

He is a University Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. His research focuses on network resource management, wireless network security, Internet of Things, 5G and beyond, and vehicular networks.

Dr. Shen received the Canadian Award for Telecommunications Research from the Canadian Society of Information Theory in 2021, the R. A. Fessenden Award in 2019 from IEEE, Canada, the Award of Merit from the Federation of Chinese Canadian Professionals (Ontario) in 2019, the James Evans Avant Garde Award in 2018 from the IEEE Vehicular Technology Society, the Joseph LoCicero Award in 2015 and the Education Award in 2017 from the IEEE Communications Society (ComSoc), and the Technical Recognition Award from Wireless Communications Technical Committee in 2019 and AHSN Technical Committee in 2013. He has also received the Excellent Graduate Supervision Award in 2006 from the University of Waterloo and the Premier's Research Excellence Award in 2003 from the Province of Ontario, Canada. He served as the Technical Program Committee Chair/Co-Chair for IEEE Globecom'16, IEEE Infocom'14, IEEE VTC'10 Fall, IEEE Globecom'07, and the Chair for the IEEE ComSoc Technical Committee on Wireless Communications. He is the President of IEEE ComSoc. He was the Vice President for Technical and Educational Activities and Publications, the Member-at-Large of the Board of Governors, the Chair of the Distinguished Lecturer Selection Committee, and the Member of the IEEE Fellow Selection Committee of ComSoc. He served as the Editor-in-Chief for the IEEE INTERNET OF THINGS JOURNAL, IEEE NETWORK, and *IET Communications*. He is a registered Professional Engineer of Ontario, Canada, a Chinese Academy of Engineering Foreign Member, a Distinguished Lecturer of the IEEE Vehicular Technology Society and Communications Society, and a Fellow of the Engineering Institute of Canada, Canadian Academy of Engineering, and Royal Society of Canada.