Digital-Twin-Empowered Resource Allocation for On-Demand Collaborative Sensing

Mushu Li, *Member, IEEE*, Jie Gao, *Senior Member, IEEE*, Conghao Zhou, *Member, IEEE*, Lian Zhao, *Fellow, IEEE*, and Xuemin Shen, *Fellow, IEEE*

***Abstract*—This article introduces an on-demand collaborative sensing scheme for industrial Internet of Things (IIoT) sensors in time-varying sensing environments, aiming to optimize the sensing performance by effectively allocating communication resources for sensory data sharing. Particularly, we propose a novel digital twins (DTs)-empowered resource allocation solution to facilitate scalable and flexible collaborative sensing. First, DTs create mathematical models using real-time network data to characterize the dynamic resource demands in collaborative sensing. Second, the performance of mathematical models in DTs is evaluated through data-driven methods. Building on our DT design, we propose a joint collaborative sensing and DT management scheme to optimize the resource allocation for sensory data sharing and DT operation. Furthermore, we develop a DT evaluation method featuring a variational autoencoder to evaluate the accuracy of DTs and enable closed-loop DT-based resource allocation. Numerical results demonstrate the effectiveness of our proposed collaborative sensing scheme in optimizing the sensing performance for all sensors.**

***Index Terms*—Collaborative sensing, digital twins (DTs), genetic algorithm, on-demand resource allocation, variational autoencoder (VAE).**

I. Introduction

INDUSTRY 5.0 is expected to facilitate customized, reliable, and resilient manufacturing processes through the enhanced collaboration between humans and machines with cutting-edge technologies, including collaborative robots (cobots) [1], [2] and cloud manufacturing [3]. Smart sensors, such as cameras on cobots, play a critical role in these new technologies, in which sensors can perceive changes in their surroundings to ensure productivity and safety in manufacturing. However, the reliability of sensing is impacted by the unpredictable dynamics of the manufacturing environment, such as human movement [4], and thus meeting the sensing performance requirements of industrial Internet of Things (IIoT) networks in Industry 5.0 presents a substantial challenge.

Collaborative sensing provides a foundational solution via enabling intelligent connections among sensors to aggregate, orchestrate, and share sensory data, which helps mitigate sensing blind spots and enhances sensing capabilities in IIoT networks [5]. The implementation of collaborative sensing has yielded significant benefits in enhancing sensing precision and facilitating greater automation in diverse Internet of Things (IoT) applications, including autonomous driving [6], [7] and environment monitoring [8]. Previous research has primarily focused on striking a balance between sensing quality and the network resource utilization associated with data aggregation and sharing. To this end, the network controller should properly identify the sensors that need additional sensory data to improve their sensing quality and allocate communication resources to support efficient sensory data exchange. In IIoT, scalability emerges as a critical metric for collaborative sensing due to the need for supporting massive sensors and low-latency industrial control. In the context of cobot applications, human motion detection must be on the order of milliseconds to effectively mitigate risks [9]. This necessitates streamlined collaborative sensing capable of promptly identifying and meeting the demands of individual sensors in a dynamic sensing environment.

Model-driven approaches for collaborative sensing tackle the scalability challenge by offline sensor selection and resource allocation based on mathematical models and statistical network information. For example, graph-theoretic methods were applied to select sensors for performing sensing and disseminating sensory data to their peers, aiming to minimize the energy consumption for sensing [10] or reduce sensory data aggregation complexity [11]. These approaches are usually explainable and applicable in a variety of scenarios but assume the known sensing quality of individual sensors and simplified (e.g., static) sensing environments. Moreover, statistical network information, such as sensing quality can be difficult to obtain, especially in highly dynamic sensing environments.

Data-driven approaches for collaborative sensing, enabled by advancements in machine learning, leverage real-time data to detect changes in the sensing environment [6] and use neural networks to learn the corresponding dynamics [12], [13]. While data-driven approaches adapt well to dynamic sensing environments, their performance can be adversely affected when the dynamics in the sensing environment are no longer

Manuscript received 12 July 2024; accepted 5 August 2024. Date of publication 22 August 2024; date of current version 20 November 2024. *(Corresponding author: Lian Zhao.)*

Mushu Li is with the Department of Computer Science and Engineering, Lehigh University, Bethlehem, PA 18015 USA (e-mail: mul224@lehigh.edu).

Jie Gao is with the School of Information Technology, Carleton University, Ottawa, ON K1S 5B6, Canada (e-mail: jie.gao6@carleton.ca).

Conghao Zhou and Xuemin Shen are with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: c89zhou@uwaterloo.ca; sshen@uwaterloo.ca).

Lian Zhao is with the Department of Electrical, Computer and Biomedical Engineering, Toronto Metropolitan University, Toronto, ON M5B 2K3, Canada (e-mail: l5zhao@torontomu.ca).

Digital Object Identifier 10.1109/JIOT.2024.3446801

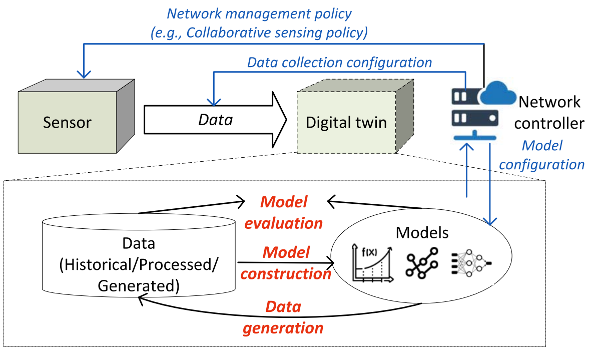

Network management policy
(e.g., Collaborative sensing policy)
Data collection configuration
Sensor
Data
Digital twin
Network controller
Model configuration
Model evaluation
Models
f(x)
Data (Historical/Processed/ Generated)
Model construction
Data generation

Fig. 1. Overview of DT-based collaborative sensing.

stationary or when handling a large number of sensors. Additionally, data-driven approaches are contingent on the regular updates of sensor status for neural network refinement, and the extensive data collection and processing can exert significant network resource consumption without a judicious design.

Digital twins (DTs), as a transformative paradigm for network management, have the potential to seamlessly integrate data-driven and model-driven approaches to empower collaborative sensing. The general idea involves creating a virtual replica for each sensor at a network controller, embedding data sets and mathematical models to capture the sensor status, operations, and demands for network resource management [14]. These models and data sets are periodically refined to reflect the changing network and sensor status [15]. The overview of DT-based collaborative sensing is presented in Fig. 1, which highlights the synergy between data and model within the DTs as follows.

1) *Model Construction From Data:* The network controller samples the status of each sensor, such as its sensing quality, and records this information in the sensor's DT. Statistical methods enable the controller to utilize the data to construct mathematical models that capture the sensor's demand for collaborative sensing, thereby providing on-demand resource allocation.
2) *Data Generation via Model:* The mathematical models within the DT allow for the prediction of future sensor statuses and the offline emulation of resource demands under varied collaborative sensing policies. This allows model-driven approaches to adapt to dynamic sensing environments.
3) *Model Evaluation via Data:* Data stored in the DT is used to assess the accuracy of the mathematical models in capturing sensor status and adapting to sensing demand dynamics, based on which the data collection strategy and the mathematical models are refined.

In this article, we propose a novel hybrid data- and model-driven solution for collaborative sensing in IIoT. Our objective is to optimize the sensing performance for all sensors by allocating communication resources for sensory data sharing while adapting to dynamic sensing environments. We achieve the objective in three phases. First, we establish a DT-based collaborative sensing framework that monitors sensor status and analyzes the sensing performance for potential collaborative sensing policies. This phase involves model construction and data generation for capturing the communication resource demand in on-demand collaborative sensing. In the second phase, we propose a joint collaborative sensing and DT management scheme that simultaneously determines the collaborative sensing policy and communication resource allocation. The scheme also optimizes the communication resource allocated to data collection for establishing DTs while balancing the overall collaborative sensing performance and the DT operation costs, aiming to enhance adaptivity and scalability of resource management for an IIoT system with a large number of sensors. Finally, we adopt a variational autoencoder (VAE) to evaluate the quality of models in DTs and predict future communication resource demand based on historical data stored in the DTs. The contributions of this article are summarized as follows.

1) Empowered by DTs, our collaborative sensing framework innovatively integrates data-driven and model-driven approaches. The framework enables on-demand collaborative sensing for a large number of sensors.
2) The proposed joint collaborative sensing and DT management scheme can optimize sensing performance while meeting the constraints of sensing collaboration delay. The performance of the proposed collaborative sensing scheme is derived in a closed form.
3) We develop a VAE-based model prediction and DT evaluation method to evaluate the quality of DTs based on the noisy DT data. The proposed method enables closed-loop resource allocation that proactively enhances sensing performance while taking into account the accuracy of DTs.

The remainder of this article is organized as follows. Section II provides an overview of related studies. Section III presents the considered system model. Sections IV and V present the DT-based collaborative sensing optimization problem and corresponding solutions, respectively. Section VI presents the simulation results, followed by the conclusion in Section VII.

II. Related Work

Collaborative sensing has been investigated in the fields of IoT and wireless sensor networks [16], [17]. Initially, the focus was on efficiently collecting and fusing sensory data from multiple sensors to capture information related to a specific spatial region [18]. With the advancement of computer vision technologies, sensors became more intelligent, capable of extracting more information, such as sensing angles [19] and occupancy of sensing blocks [20], from complex sensory data, such as LiDAR images [20], [21]. Correspondingly, the target of collaborative sensing shifted from maximizing sensing coverage to improving the quality of sensory data within the area of interest. To efficiently utilize the communication resources in sensory data exchange, the sensory data from multiple sensors is aggregated according to their sensing quality, and the result is delivered to the sensors in need.

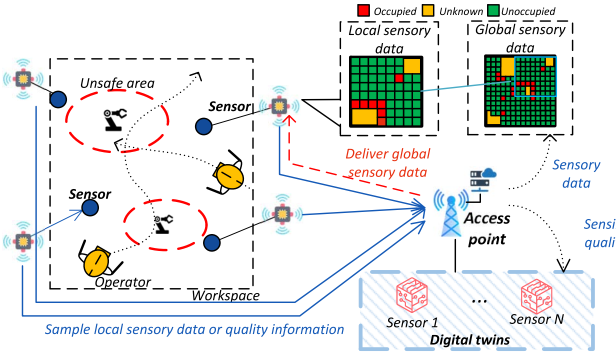


Fig. 2. System scenario for collaborative sensing.

However, such a collaborative sensing pattern faces challenges due to the dynamic nature of the sensing environment and the corresponding sensory data captured by each sensor. To address this challenge, reinforcement learning methods were adopted to select sensors that can capture useful sensory data to participate in collaborative sensing, aiming to maximize the overall sensing quality while minimizing resource consumption in the long term [6], [22], [23]. In IIoT, the massive scale of sensors necessitates prompt collaborative sensing decisions with low signaling overhead [24]. While machine learning has shown promise in solving complex collaborative sensing problems, pure data-driven approaches cannot provide insights regarding sensor-level demands. Moreover, the data-driven approaches may not perform well in scenarios where the number of active sensors varies over time and network environments change frequently [14], [25].

The DT technology provides a framework for the synergy of data-driven and model-driven approaches in network resource management. In our visionary paper [14], we envisioned a DT-based resource allocation architecture, wherein DTs are established for individual devices and network slices. These DTs capture device-specific service demands and enable the emulation of networking policies, leading to proactive network performance optimization. Utilizing the network status and demand data models provided by DTs, a network controller can tailor resource allocation methods to precisely meet the quality of service requirements for each device [26] and further improve the quality-of-experience for network users [27]. In [28], DTs were created to simulate complex and changing network environments in IIoT and facilitate reinforcement learning for adaptive resource allocation. In our paper, we answer the question of how to integrate the data-driven and model-driven approaches to build precise sensor DTs for optimizing collaborative sensing performance.

III. System Model

The considered IIoT collaborative sensing scenario is illustrated in Fig. 2. There is a single access point (AP) equipped with a network controller to cover a limited area, e.g., the workspace of a manufacturing facility. The entire area can be divided into $S_X \times S_Y$ subareas, and the set of all subareas is denoted by $\mathcal{L}$. A set of sensors, denoted by $\mathcal{N} = \{1, \ldots, N\}$, are deployed in the area. Each sensor (e.g., a camera or radar) covers a part of the area and connects with the AP via a wireless link. The set of subareas covered by sensor n is denoted by $\mathcal{L}_n$. The sensing coverage areas of different sensors may overlap, and the number of subareas covered by a sensor is a constant denoted by L. We adopt time-slotted sensing and denote the slot index and the slot duration by t and τ, respectively. In each time slot, every sensor senses and perceives its surrounding environment to safeguard the operations of human workers and IIoT devices, such as cobots.

The sensory data captured by each sensor is referred to as *local sensory data*. The AP aggregates local sensory data to create a comprehensive sensory data set for the entire area, referred to as *global sensory data*. Due to the obstacles in the physical environment and the movements of devices and workers, the sensors may not always successfully sense all the covered subareas, and the local sensory data captured by individual sensors may be incomplete. To address this issue, the AP identifies sensors with low sensing quality and delivers the corresponding parts of global sensory data to them.[1] If a sensor receives the global sensory data, the sensor merges the data with its own local sensory data to potentially enhance sensing quality; the resulting sensory data of the sensor is referred to as the sensor's *final sensory data*. To keep the global sensory data up-to-date, the AP sequentially prompts the sensors to upload their local sensory data and then merges the received data with the global sensory data successively.

In time slot t, the quality of local sensory data for subarea l captured by sensor n is denoted by $Q_{n,t,l}$, where $Q_{n,t,l} \in \{0, 1\}$. The value of $Q_{n,t,l}$ equals one if sensor n can successfully sense subarea l without blockage, and the value equals zero otherwise. The local sensing quality of sensor n in time slot t is defined as the average quality of its local sensory data over all subareas in $\mathcal{L}_n$, denoted by $\bar{Q}_{n,t}$. In addition, the quality of the global sensory data and the final sensory data at sensor n for subarea l in time slot t are denoted by $Q_{G,t,l}$ and $G_{n,t,l}$, respectively. Similarly, the global sensing quality in time slot t is defined as the average quality of the global sensory data at the corresponding time slot over all subareas in $\mathcal{L}$, denoted by $\bar{Q}_{G,t}$; the final sensing quality of sensor n in time slot t is defined as the average quality of the final sensory data over all subareas in $\mathcal{L}_n$, denoted by $\bar{G}_{n,t}$. The sizes of the local sensory data generated by any sensor and the part of global sensory data within any sensor's sensing coverage are a constant, denoted by V.

To identify which sensors to receive the global sensory data, we adopt a threshold-based collaborative sensing policy defined as follows.

Definition 1 (Threshold-Based Collaborative Sensing): The AP generates a threshold denoted by ρ. The AP delivers the corresponding part of the global sensory data to the sensor if the sensor's current local sensing quality is below the threshold, i.e., $\bar{Q}_{n,t} < \rho$.

Using a threshold allows the AP to quickly identify targets in collaborative sensing when managing a large number of sensors in the IIoT scenario. By selecting an appropriate

[1] Unicast transmission is used to deliver a part of the global sensory data to each identified sensor.

threshold, the overall sensing quality across all sensors can be maximized subject to the constraints on communication resources, when sensors are evenly distributed throughout the entire area. The threshold needs to be adjusted periodically to adapt to the dynamics in the sensing environment. The interval between time slots when the threshold is adjusted is considered a decision window, and we denote the set of time slots within decision window k as $\mathcal{W}_k$. The threshold for collaborative sensing applied to decision window k is denoted by ρ_k.

Due to the large number of sensors, it is difficult to have each sensor update its sensing quality information to the AP in every time slot for collaborative sensing. The AP maintains a DT for each sensor to monitor the sensor status and estimate its sensing quality in each time slot. This approach allows the AP to identify sensors with low sensing quality and deliver the corresponding global sensory data to them. The sensing quality of sensor n estimated by its DT is denoted by $\hat{Q}_{n,t}$. Additionally, the DT of the sensor also analyzes the sensor status for resource allocation, to be detailed in Section III-C. To accurately monitor the sensor status, the AP may collect the sensors' real-time sensing quality to update the DTs. Therefore, communication resources should be properly allocated to sensors for uploading the real-time quality of local sensory data, referred to as *quality status*, in addition to aggregating local sensory data and delivering global sensory data.

A. Sensing Quality Model

Next, we model the global sensing quality achieved by aggregating local sensory data. The AP continuously prompts sensors to upload their local sensory data and updates the received local sensory data for subareas, which has higher quality than the current global sensory data for those subareas, into the global sensory data. Specifically, when the prompted sensor successfully senses subarea l, the global sensory data for that subarea is updated, increasing its corresponding quality. In contrast, if the prompted sensor fails to sense subarea l or if subarea l is not in the sensor's sensing coverage, the existing global sensory data is reused, and the corresponding quality of the subarea degrades over time. We denote the set of sensors whose local sensory data is received and processed by the AP in time slot t as $\mathbf{u}_t$, as multiple sensors may upload their local sensory data to the AP within a time slot. As a result, the evolution of the quality of the global sensory data for subarea l over time slots can be modeled as follows:

$$Q_{G,t+1,l} = \begin{cases} \eta^{-\tau} Q_{G,t,l}, \quad \text{if } l \notin \{\cup_{n\in\mathbf{u}_t}\mathcal{L}_n\} \\ \max\left\{\eta^{-\tau} Q_{G,t,l}, \{\eta^{-h\tau} Q_{n,t-h,l} \quad \forall n \in \mathbf{u}_t\}\right\} \\ \text{if } l \in \{\cup_{n\in\mathbf{u}_t}\mathcal{L}_n\} \end{cases} \quad (1)$$

where η represents a degradation parameter and h is the number of time slots elapsed from the moment when the selected sensor n uploaded its local sensory data till time slot $t+1$.

Based on the threshold-based collaborative sensing policy in Definition 1, we then model the final sensing quality of the sensor with collaborative sensing. In each time slot, the AP estimates the local sensing quality of each sensor using the DTs. If the estimated local sensing quality of the sensor is greater than threshold ρ_k, the AP will not deliver the global sensory data to the sensor. The processing time for sensors to analyze their local sensory data is considered to be one time slot. In such a case, the quality of the final sensory data for subarea l at sensor n in time slot t is

$$G_{n,t,l} = \eta^{-\tau} Q_{n,t,l}, \text{ if } \hat{Q}_{n,t} \geq \rho_k. \quad (2)$$

Otherwise, the sensor receives the global sensory data from the AP. The sensor keeps the received global sensory data in a subarea, which has a higher quality compared to the corresponding local sensory data, as the final sensory data for the subarea. To ensure that the global sensory data can be timely delivered, we consider a delay constraint that the delay for delivering the global sensory data to the corresponding sensor cannot exceed τ, i.e., the duration of a time slot. Thus, the final sensing quality of sensor n for subarea l in time slot t can be estimated as

$$G_{n,t,l} = \max\left\{\eta^{-\tau} Q_{n,t,l}, \eta^{-\tau} Q_{G,t,l}\right\}, \text{ if } \hat{Q}_{n,t} < \rho_k. \quad (3)$$

B. DT Model

A higher threshold for collaborative sensing enables more sensors to receive global sensory data, potentially enhancing the sensing quality. However, delivering global sensory data to more sensors within the specified latency could exceed the available bandwidth. On-demand collaborative sensing is necessary for maximizing resource utilization. Therefore, we establish a DT for each sensor to monitor its real-time local sensing quality, capture the sensing environment dynamics, and identify the resource demands of the sensor in collaborative sensing.

Denote the DT of sensor n in time slot t by $\{\mathcal{I}_{n,t}, \mathcal{M}_{n,t}, \mathcal{D}_{n,t}, \mathcal{O}_{n,t}\}$, where the four elements represent the set of inputs, mathematical models, the data set, and outputs in DT n. The DT model is illustrated in Fig. 3(a) and elaborated as follows:

1) Inputs $\mathcal{I}_{n,t}$: The input to the sensor's DT is the quality status. The DT of a sensor can obtain such information of the sensor from the uploaded local sensory data. In addition, the AP also occasionally collects the quality status from the sensor directly. We define the number of time slots between two successive updates of the quality status of the sensor, due to the upload of either local sensory data or quality status, as an *update interval*, denoted by T_k. The length of the update interval depends on the communication resources allocated to uplink transmissions. A short update interval enables the DT to accurately capture the sensor status.

2) Mathematical Models $\mathcal{M}_{n,t}$: In the DTs, mathematical models are generated from four functions: 1) monitoring; 2) abstraction; 3) prediction; and 4) evaluation. The models from these four functions are represented as $\mathcal{M}_n^{\mathrm{M}}$, $\mathcal{M}_n^{\mathrm{A}}$, $\mathcal{M}_n^{\mathrm{P}}$, and $\mathcal{M}_n^{\mathrm{E}}$, respectively. The parameters of the models are updated periodically according to the inputs and the data sets of DTs.[2]

Monitoring: The status monitoring function uses a mathematical model, i.e., $\mathcal{M}_n^{\mathrm{M}}$, to estimate the local sensing quality

[2] We assume that the types of mathematical models are determined in advance for simplicity.

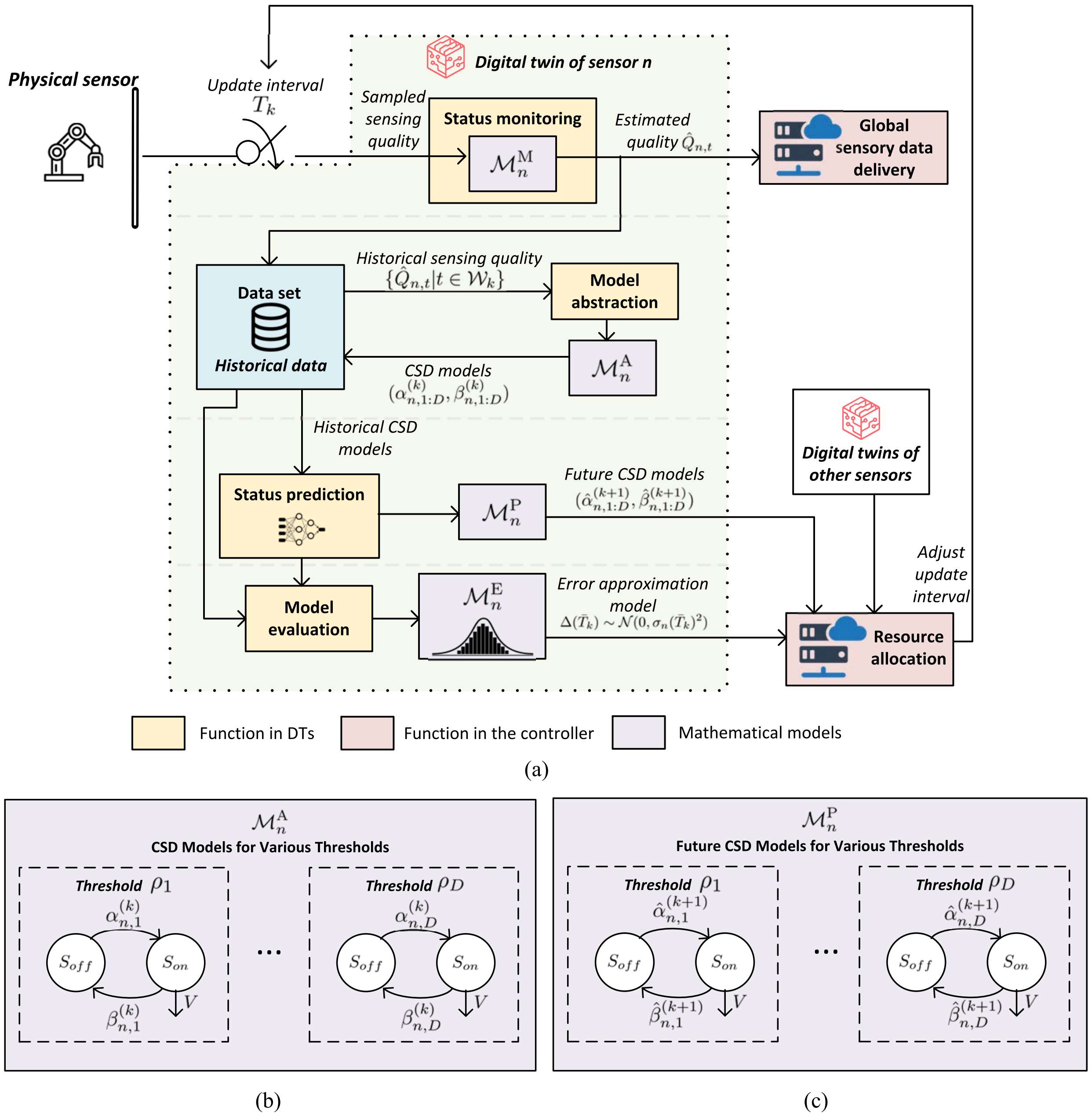

Physical sensor
Update interval T_k
Digital twin of sensor n
Sampled sensing quality
Status monitoring
$\mathcal{M}_n^{\mathrm{M}}$
Estimated quality $\hat{Q}_{n,t}$
Global sensory data delivery
Data set
Historical data
Historical sensing quality $\{\hat{Q}_{n,t}|t \in \mathcal{W}_k\}$
Model abstraction
$\mathcal{M}_n^{\mathrm{A}}$
CSD models $(\alpha_{n,1:D}^{(k)}, \beta_{n,1:D}^{(k)})$
Historical CSD models
Status prediction
$\mathcal{M}_n^{\mathrm{P}}$
Future CSD models $(\hat{\alpha}_{n,1:D}^{(k+1)}, \hat{\beta}_{n,1:D}^{(k+1)})$
Digital twins of other sensors
Adjust update interval
Model evaluation
$\mathcal{M}_n^{\mathrm{E}}$
Error approximation model $\Delta(\bar{T}_k) \sim \mathcal{N}(0, \sigma_n(\bar{T}_k)^2)$
Resource allocation
Function in DTs
Function in the controller
Mathematical models
(a)
$\mathcal{M}_n^{\mathrm{A}}$
CSD Models for Various Thresholds
Threshold ρ_1
$\alpha_{n,1}^{(k)}$
S_{off}
S_{on}
$\beta_{n,1}^{(k)}$
V
...
Threshold ρ_D
$\alpha_{n,D}^{(k)}$
$\beta_{n,D}^{(k)}$
(b)
$\mathcal{M}_n^{\mathrm{P}}$
Future CSD Models for Various Thresholds
Threshold ρ_1
$\hat{\alpha}_{n,1}^{(k+1)}$
$\hat{\beta}_{n,1}^{(k+1)}$
Threshold ρ_D
$\hat{\alpha}_{n,D}^{(k+1)}$
$\hat{\beta}_{n,D}^{(k+1)}$
(c)

Fig. 3. Overview of the DT for a sensor. (a) DT structure. (b) CSD models of sensor n in time window k. (c) Future CSD models of sensor n for time window $k+1$.

of the sensor at time slot n, i.e., $\hat{Q}_{n,t}$, based on the historical sample inputs to the DT, i.e., the local sensing quality status. The mathematical model can be a regression model, a Kalman filter, or other data exploitation methods. We assume the model is given and pretrained, while the estimation is imperfect due to the nonzero update interval.

Abstraction: Moving objects and obstacles in the sensing environment result in time-varying sensing quality among subareas. Therefore, instead of modeling each moving object, we model the collaborative sensing demand (CSD) dynamics of individual sensors, which reflect the sensing quality fluctuations, for a lower complexity. In the DT of sensor n, the abstraction function updates the mathematical model $\mathcal{M}_n^{\mathrm{A}}$ as illustrated in Fig. 3(b). Model $\mathcal{M}_n^{\mathrm{A}}$ involves a set of two-stage on-off Markov models to approximate the demand for receiving global sensory data from the AP. The settings of the on-off models in $\mathcal{M}_n^{\mathrm{A}}$ are as follows: Each DT discretizes the continuous threshold into D levels, resulting in a set of threshold candidates $\boldsymbol{\rho} = \{\rho^1, \ldots, \rho^D\}$, where $\rho^1 = 0$ and $\rho^D = 1$. For each threshold $\rho^d \in \boldsymbol{\rho}$, an two-stage on-off Markov model is established, referred to as the *CSD model* corresponding to threshold ρ^d. States S_{on} and S_{off} represent the estimated quality $\hat{Q}_{n,t}$ are less than and greater than threshold ρ^d, respectively. When the state is S_{on}, the AP should deliver the corresponding global sensory data to the sensor at V bits per time slot, i.e., the size of the corresponding global sensory data. The transition rates from S_{off} to S_{on} and from S_{on} to S_{off}, denoted by $\alpha_{n,d}^{(k)}$ and $\beta_{n,d}^{(k)}$, respectively, are updated periodically at the end of each decision window as the parameters in model $\mathcal{M}_n^{\mathrm{A}}$. The term $(\alpha_{n,1:D}^{(k)}, \beta_{n,1:D}^{(k)})$ represents the pair of transition rates for all thresholds $\boldsymbol{\rho}$ at decision window k. The transition rates for each threshold candidate can be estimated empirically by counting the number of transitions via comparing the estimated sensing quality, i.e., $\{\hat{Q}_{n,t} | t \in \mathcal{W}_k\}$, from the data. The abstracted CSD models can reveal the communication resources that would be consumed for collaborative sensing with different thresholds in the preceding decision window.

Status Prediction: We use a VAE, to be detailed in Section V-C, as a status prediction function to update the mathematical model $\mathcal{M}_n^{\mathrm{D}}$. Model $\mathcal{M}_n^{\mathrm{E}}$ involves a set of predicted CSD models in the upcoming decision window and is referred to as *future CSD models*. The structure of $\mathcal{M}_n^{\mathrm{E}}$ is similar with model $\mathcal{M}_n^{\mathrm{A}}$, as illustrated in Fig. 3(c), while the transition rates of $\mathcal{M}_n^{\mathrm{E}}$ are predicted from the historical transition rates of CSD models periodically. Specifically, at the beginning of decision window $k+1$, the transition rates in the future CSD models, denoted by $(\hat{\alpha}_{n,1:D}^{(k+1)}, \hat{\beta}_{n,1:D}^{(k+1)})$, are predicted according to the CSD models in the W preceding decision windows, i.e., $\{(\alpha_{n,1:D}^{(k-W+1)}, \beta_{n,1:D}^{(k-W+1)}), \ldots, (\alpha_{n,1:D}^{(k)}, \beta_{n,1:D}^{(k)})\}$. The future CSD models are used by the network controller to allocate the communication resources for the upcoming time window proactively.

Evaluation: The aforementioned CSD models are constructed based on the estimated local sensing quality which is provided by the state monitoring function. Note that these estimates may deviate from the ground truth. The model evaluation function updates the mathematical model $\mathcal{M}_n^{\mathrm{E}}$ to assess the distribution of estimation errors, represented as an unbiased Gaussian distribution, i.e., $\Delta(\bar{T}_k) \sim \mathcal{N}(0, \sigma_n(\bar{T}_k)^2)$.[3] The standard deviation, $\sigma_n(\cdot)$, is a nondecreasing function of the update interval, parameterized by a set of parameters $\mathcal{K}_n$. This evaluation function reuses the VAE from the status prediction function and deduces $\mathcal{K}_n$ by comparing the generated CSD models across various update intervals. The details are presented in Section V-C. Model $\mathcal{M}_n^{\mathrm{E}}$ enables the AP to allocate communication resources and efficiently update the DTs.

3) Data Set $\mathcal{D}_{n,t}$: A data set is adopted at each DT to store the data of the corresponding sensor. The data in a DT includes the estimated sensing quality of the sensor, corresponding updating intervals, and model parameters, including the transition rates of the CSD models in the preceding W decision windows. The data set is continuously updated over time slots for recording the real-time state of the sensor, allowing the functions in the DT to capture the changes in the CSD models and refine the models accordingly.

4) Output $\mathcal{O}_{n,t}$: Each DT provides three outputs to the network controller for resource allocation. The first output is the estimated local sensing quality of the sensor in current time slot n, i.e., $\hat{Q}_{n,t}$, from the status monitoring function in the DT. Based on this, the network controller determines whether to deliver the global sensory data to the sensor. The second output is the predicted future CSD models, i.e., $(\hat{\alpha}_{n,1:D}^{(k+1)}, \hat{\beta}_{n,1:D}^{(k+1)})$, at the beginning of each decision window. The third output is the estimation error distribution $\Delta(\bar{T}_k)$. The network controller gathers these outputs from DTs to allocate communication resources for sensory data collection and delivery, as well as to adjust the update interval for the DTs.

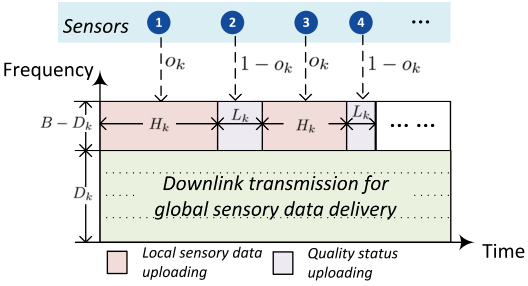


Fig. 4. Illustration of the communication bandwidth allocation.

C. Communication and Computing Model

Communication resources need to be allocated for three types of traffic: 1) downlink transmissions for global sensory data delivery; 2) uplink transmissions for local sensory data aggregation; and 3) uplink transmissions for uploading quality status. The communication bandwidth allocation model is illustrated in Fig. 4. The overall communication bandwidth, denoted by B, is divided for uplink and downlink transmissions. In decision window k, the bandwidths allocated for downlink and uplink transmissions are denoted by D_k and $B - D_k$, respectively. The latter is further divided into two different types of uplink transmissions.

1) Downlink Transmission: Sufficient bandwidth should be allocated to global sensory data delivery to ensure that the global sensory data can be delivered in a time slot. Given the downlink bandwidth D_k, the data rate that can be achieved for global sensory data delivery, denoted by $R_{D,k}$, is

$$R_{D,k} = D_k \log\left(1 + \frac{S_D}{D_k N_0}\right) \tag{4}$$

where S_D denotes the received communication power at the sensor and N_0 represents the noise power spectral density. The received communication power is identical for all sensors via proper power control. The AP has a transmission buffer to store the data that is to be transmitted to the sensors. The state of the transmission buffer, i.e., the number of bits in the buffer, in time slot t, is denoted by μ_t. According to the threshold in collaborative sensing ρ_k, the evolution of the buffer state can be formulated as

$$\mu_{t+1} = \max\{\mu_t - \tau R_{D,k}, 0\} + V \sum_{n \in \mathcal{N}} \mathbf{1}\left[\hat{Q}_{n,t} < \rho_k\right] \tag{5}$$

where the function $\mathbf{1}[x]$ is one if x is true and zero otherwise. The buffer state μ_t should always be lower than the maximum number of bits in the buffer $\mu_{\max}$ with at least probability ϱ to ensure that the global sensory data can be delivered on time. To ensure that the global sensory data is delivered to a sensor within a time slot, we have $\mu_{\max} = \tau R_{D,k} - V$. We assume that the sensing environment is similar among all

[3] Note that, the distribution of the unbiased estimation error in the proposed solution is not limited to the Gaussian distribution.

sensors within the considered area during a decision window. Under this assumption, the buffer overflow probability can be obtained based on the average transition rates of the CSD models over the sensors, provided by DTs. Specifically, let $\bar{\alpha}_d^{(k)} = (1/N)\sum_n \hat{\alpha}_{n,d}^{(k)}$ and $\bar{\beta}_d^{(k)} = (1/N)\sum_n \hat{\beta}_{n,d}^{(k)}$ for all threshold $\rho^d \in \boldsymbol{\rho}$. The utilization factor of allocated downlink bandwidth is

$$\varpi = \left(\frac{\bar{\alpha}_d^{(k)}}{\bar{\alpha}_d^{(k)} + \bar{\beta}_d^{(k)}}\right)\frac{N}{C_k} < 1 \tag{6}$$

where C_k represents the number of sensors to receive the global sensory data in a time slot and $C_k = (\tau R_{d,k}/V)$. The probability of buffer state exceeding $\mu_{\max}$ is given by [29]

$$P(l > \mu_{\max}) \approx A_N \varpi^N e^{-r\bar{\beta}_d^{(k)}\mu_{\max}/V} \tag{7}$$

where parameter r is defined by

$$r = \frac{(1-\varpi)\left(1 + \bar{\alpha}_d^{(k)}/\bar{\beta}_d^{(k)}\right)}{1 - C_k/N}$$

and parameter A_N is defined by

$$A_N = \Pi_{i=1}^{N-\lfloor C_k\rfloor - 1}\frac{z_n}{z_n + r}. \tag{8}$$

In (8), z_n is the nth eigenvalue of a matrix $F = \mathbf{M}\mathbf{D}^{-1}$, where $\mathbf{M}$ is an $(N+1)\times(N+1)$ matrix and the nonzero elements are defined as follows:

$$\begin{aligned}
\mathbf{M}[i,i] &= -\left[i\bar{\beta}_{d,k} + (N-i)\bar{\alpha}_d^{(k)}\right], i \in \{0,\ldots,N\}\\
\mathbf{M}[i-1,i] &= i\bar{\beta}_{d,k}, i \in \{1,\ldots,N\}\\
\mathbf{M}[i,i+1] &= (N-i)\bar{\alpha}_d^{(k)}, i \in \{0,\ldots,N-1\}.
\end{aligned}$$

Furthermore, $\mathbf{D}$ is defined as follows:

$$\mathbf{D} = \operatorname{diag}\left[-C_k\bar{\beta}_{d,k}, (1-C_k)\bar{\beta}_{d,k}, \ldots, (N-C_k)\bar{\beta}_{d,k}\right]. \tag{9}$$

The derivation is given in [29] and omitted here. A proper threshold and downlink bandwidth should be selected to ensure the buffer overflow probability is within acceptable limits.

2) Uplink Transmissions: We consider a time-division model for each sensor to upload its status data, either local sensory data or its quality status, to the AP. Specifically, the AP randomly selects a sensor to upload its status data based on a uniform distribution, where each sensor have probability $1/N$ to upload their data. After the sensor is selected, with probability o_k, the selected sensor uploads its current local sensory data for updating the global sensory data at the AP and its DT. With probability $1 - o_k$, it uploads the current quality status for updating its DT only. After the AP receives the status data from a sensor, it processes the status data, and the processing time follows exponential distributions. Specifically, the expected processing rates for processing local sensory data and the quality status are constants and denoted by λ_H and λ_L, respectively. Thus, the expected delays for uploading the local sensory data, denoted by $\mathbb{E}[H_k]$, and for updating the quality status, denoted by $\mathbb{E}[L_k]$, are

$$\begin{cases}\mathbb{E}[H_k] = V\left(\frac{1}{R_{U,k}} + \frac{1}{\lambda_H}\right)\\ \mathbb{E}[L_k] = V_L\left(\frac{1}{R_{U,k}} + \frac{1}{\lambda_L}\right)\end{cases} \tag{10}$$

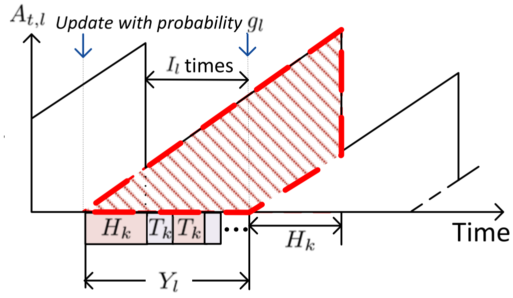


Fig. 5. AoI evolution of the global sensory data for subarea l.

respectively. In (10), $R_{U,k}$ is the uplink transmission data rate and can be defined by

$$R_{U,k} = (B - D_k)\log\left(1 + \frac{S_U}{(B-D_k)N_0}\right) \tag{11}$$

where S_U denotes the received communication power at the AP. Furthermore, V_L represents the data size of the quality status, and $V_L < V$. Then, we have the expected update interval as follows:

$$\bar{T}_k = o_k\mathbb{E}[H_k]N + (1-o_k)\mathbb{E}[L_k]N. \tag{12}$$

According to (12), frequently sending the quality status, i.e., choosing a small o_k, can reduce the update interval and improve the accuracy of the DT at the cost of a longer interval for updating the global sensory data. The quality of the global sensory data depends on the time elapsed since the latest update of the local sensory data for each subarea, i.e., the Age of Information (AoI) of the global sensory data.

IV. DT-Enabled Performance Evaluation and Problem Formulation

In this section, we first evaluate the final sensing quality of sensors using the CSD models from their DTs. Then, we formulate the optimization problem of maximizing the final sensing quality.

A. Performance Evaluation

In each time slot, the final sensing quality of a sensor depends on the quality of both the local sensory data and the received global sensory data. The quality of global sensory data for a subarea evolves over time, which can be captured by the AoI of the global sensory data. Let $A_{t,l}$ and $\bar{A}_l$ represent the AoI of the global sensory data for subarea l in time slot t and the expected AoI of the global sensory data for subarea l over time, respectively. The expected quality of global sensory data in subarea l over time slots, denoted by $\check{Q}_{G,l}$, can be represented as

$$\check{Q}_{G,l} = \mathbb{E}\left[\eta^{A_{t,l}}\right] \geq \eta^{\bar{A}_l}. \tag{13}$$

The inequality is achieved due to the convexity of $\eta^{A_{t,l}}$ on $A_{t,l}$ when $0 < \eta$. We use the lower bound to analyze the sensing quality, i.e., $\check{Q}_{G,l} = \eta^{\bar{A}_l}$. The evolution of AoI regarding the global sensory data for subarea l is illustrated in Fig. 5. Let Y_l

be a random variable representing the interval between events when a sensor senses and updates the sensory data for subarea l to the AP. The expected AoI of the global sensory data for subarea l can be derived as [30]

$$\begin{aligned}\bar{A}_l &= \left\{\frac{1}{2}\mathbb{E}\left[(Y_l + H_k)^2\right] - \frac{1}{2}\mathbb{E}\left[H_k^2\right]\right\}/\mathbb{E}[Y_l]\\ &= \left\{\mathbb{E}[Y_l H_k] + \frac{1}{2}\mathbb{E}\left[Y_l^2\right]\right\}/\mathbb{E}[Y_l]. \end{aligned}\tag{14}$$

The numerator in (14) represents the average value of the shadowed area in Fig. 5.

The interval Y_l depends on the number of times that sensors upload data to the AP within this interval and the update interval T_k. Let I_l denote the number of times that sensors upload their status data to the AP within the interval Y_l after a sensor senses and updates local sensory data for subarea l. The value of I_l depends on the probability that the local sensory data for subarea l is updated by a sensor, which is denoted by g_l. According to the communication and computing model in Section III, g_l can be obtained as

$$g_l = \kappa \times o_k |\mathcal{N}_{c,l}|/N, \text{ where } \mathcal{N}_{c,l} = \{n | l \in \mathcal{L}_n\} \tag{15}$$

where parameter κ represents the average local sensing quality for a subarea and can be derived from the statistics of historical local sensing quality. Accordingly, I_l follows a geometric distribution, where the probability mass function (PMF) is

$$P(I_l = i) \ = (1 - g_l)^i g_l, i \geq 0. \tag{16}$$

From Fig. 5, the interval Y_l includes the time that a sensor senses subarea l to update its local sensory data, i.e., H_k, and the interval that other sensors update their status data except for updating the sensory data for subarea l, i.e., $I_l \times T_k$. Therefore, the mean of Y_l is

$$\mathbb{E}[Y_l] = \mathbb{E}[H_k] + \mathbb{E}[I_l T_k]. \tag{17}$$

In addition, considering the independence between update interval T_k and the update times I_l, we can obtain $\mathbb{E}[Y_l H_k]$ in (14) as

$$\mathbb{E}[Y_l H_k] = \mathbb{E}\left[H_k^2\right] + \mathbb{E}[H_k T_k]\mathbb{E}[I_l]$$

where $\mathbb{E}[H_k T_k]$ can be rewritten from evaluating the conditions of T_k

$$\begin{aligned}\mathbb{E}[H_k T_k] &= o_k \mathbb{E}[H_k T_k | T_k = H_k] + (1 - o_k)\mathbb{E}[H_k T_k | T_k = L_k]\\ &= o_k \mathbb{E}\left[H_k^2\right] + (1 - o_k)\mathbb{E}[H_k]\mathbb{E}[L_k]. \end{aligned}\tag{18}$$

Similarly, $\mathbb{E}[Y_l^2]$ in (14) can be obtained as follows:

$$\mathbb{E}\left[Y_l^2\right] = \mathbb{E}\left[H_k^2\right] + 2\mathbb{E}[H_k T_k]\mathbb{E}[I_l] + \mathbb{E}\left[I_l^2\right]\mathbb{E}\left[T_k^2\right].$$

Thus, the expected AoI of the global sensory data for subarea l is

$$\bar{A}_l = \frac{2\mathbb{E}[I_l]\mathbb{E}[H_k T_k] + \frac{3}{2}\mathbb{E}\left[Y_l^2\right] + \frac{1}{2}\mathbb{E}\left[I_l^2\right]\mathbb{E}\left[T_k^2\right]}{\mathbb{E}[H_k] + \mathbb{E}[I_l]\bar{T}_k}. \tag{19}$$

The AoI in (19) can be used to determine the quality of the global sensory data for a subarea in (13).

The quality of final sensory data for subarea l at sensor n, i.e., $G_{n,t,l}$, depends on the quality of both local sensory data and received global sensory data from the AP of the sensor. From (1)–(3), in each time slot, there are four possible cases in determining the quality of final sensory data as follows.

1) *Case 1:* No global sensory data is received at the sensor since its local sensing quality exceeds the threshold for collaborative sensing, i.e., $\bar{Q}_{n,t} \geq \rho_k$. In this case, the quality of the final sensory data for subarea l is the same as that of the local sensory data, i.e., $G_{n,t,l} = \eta^{-\tau} Q_{n,t,l}$;
2) *Case 2:* The sensor receives the global sensory data for subarea l, and the corresponding global sensory data is obtained from the local sensory data updated by other sensors within one time slot. The condition of such a case can be estimated as $I_l \bar{T}_k + \mathbb{E}[H_k] \leq \tau$, and the quality of the final sensory data for subarea l is the same as that of the received global sensory data.
3) *Case 3:* The sensor receives global sensory data and successfully senses subarea l. In addition, the received global sensory data cannot be obtained within one time slot, i.e., $I_l \bar{T}_k + \mathbb{E}[H_k] > \tau$. The quality of final sensory data for subarea l is the same as that of the corresponding local sensory data.
4) *Case 4:* The sensor receives global sensory data and cannot successfully sense subarea l due to the blockage. In addition, the received global sensory data cannot be obtained within one time slot, i.e., $I_l \bar{T}_k + \mathbb{E}[H_k] \geq \tau$. The quality of the final sensory data for subarea l is the same as that of the corresponding global sensory data.

Denote the expected quality of the final sensory data for subarea l at a sensor over time by $\check{G}_l$. According to the above four cases, $\check{G}_l$ can be determined as follows:

$$\check{G}_l = \frac{1}{N|\mathcal{W}_k|}\sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{W}_k} G_{n,t,l} = \sum_{c=1}^{4} P_{l,c}^{G}\eta^{-L_{l,c}} \tag{20}$$

where $c \in \{1, 2, 3, 4\}$ is the index of the case (i.e., one of the aforementioned four cases) and $L_{l,c}$ represents the expected AoI of the final sensory data. In (20), $P_{l,c}^{G}$ represents the probability of case c occurring in a time slot. In terms of the expected AoI of the final sensory data, for the first and fourth cases, the local sensory data is used. Thus, the corresponding expected AoI, i.e., $L_{l,1}$ and $L_{l,4}$, are equal to the length of a time slot τ. The expected AoI in the second and third cases, i.e., $L_{l,2}$ and $L_{l,3}$, depend on the condition whether $I_l \bar{T}_k + \mathbb{E}[H_k]$ is greater or less than τ. Let $\hat{I} = (\tau - \mathbb{E}[H_k])/\bar{T}_k$. The value of $L_{l,2}$ and $L_{l,3}$ can be obtained from

$$L_{l,2} = \frac{2\mathbb{E}\left[I_l | I_l \leq \hat{I}\right]\mathbb{E}[H_k T_k] + \frac{3}{2}\mathbb{E}\left[Y_l^2\right] + \frac{1}{2}\mathbb{E}\left[I_l^2 | I_l \leq \hat{I}\right]\mathbb{E}\left[T_k^2\right]}{\mathbb{E}[H_k] + \mathbb{E}\left[I_l | I_l \leq \hat{I}\right]\bar{T}_k} \tag{21a}$$

$$L_{l,3} = \frac{2\mathbb{E}\left[I_l | I_l > \hat{I}\right]\mathbb{E}[H_k T_k] + \frac{3}{2}\mathbb{E}\left[Y_l^2\right] + \frac{1}{2}\mathbb{E}\left[I_l^2 | I_l > \hat{I}\right]\mathbb{E}\left[T_k^2\right]}{\mathbb{E}[H_k] + \mathbb{E}\left[I_l | I_l > \hat{I}\right]\bar{T}_k} \tag{21b}$$

respectively. In (21a) and (21b), the mean and second moment of I_l with the conditions $I_l \leq \hat{I}$ and $I_l > \hat{I}$ are derived, respectively, in Appendix A. Furthermore, given threshold for collaborative sensing ρ^d, $P^G_{l,1}$ through $P^G_{l,4}$ can be formulated as follows:

$$P^G_{l,1} = \frac{\kappa \bar{\beta}_d^{(k)}}{\bar{\alpha}_d^{(k)} + \bar{\beta}_d^{(k)}};\ P^G_{l,2} = \frac{P\left(I_l \leq \hat{I}\right)\bar{\alpha}_d^{(k)}}{\bar{\alpha}_d^{(k)} + \bar{\beta}_d^{(k)}} \tag{22a}$$

$$P^G_{l,3} = \frac{(1-\kappa)P\left(I_l > \hat{I}\right)\bar{\alpha}_d^{(k)}}{\bar{\alpha}_d^{(k)} + \bar{\beta}_d^{(k)}};\ P^G_{l,4} = \frac{\kappa P\left(I_l > \hat{I}\right)\bar{\alpha}_d^{(k)}}{\bar{\alpha}_d^{(k)} + \bar{\beta}_d^{(k)}}. \tag{22b}$$

By (20), given a resource allocation strategy, the network controller can use the predicted CSD models from DTs to evaluate the final sensing quality of all sensors with different thresholds for collaborative sensing.

B. Problem Formulation

We aim to maximize the final sensing quality achieved by collaborative sensing for sensors, where both the threshold for collaborative sensing, i.e., threshold ρ_k, and the corresponding communication resource allocation need to be jointly determined. For decision window k, the optimization problem can be formulated as follows:

$$[\mathbf{P}]\ \max_{\substack{\rho_k, D_k,\\ o_k}}\ \frac{1}{NL|\mathcal{W}_k|}\sum_{t\in\mathcal{W}_k}\sum_{n\in\mathcal{N}}\sum_{l\in\mathcal{L}_n} G_{n,t,l} \tag{23a}$$

$$\text{s.t.}\ P(\mu_t < \mu_{\max}) \geq \varrho \tag{23b}$$

$$D_k \leq B, 0 \leq o_k \leq 1 \tag{23c}$$

$$\rho_k \in \left[\rho^1, \ldots, \rho^D\right]. \tag{23d}$$

The objective of $\mathbf{P}$ is to maximize the expected final sensing quality for all sensors. The optimization problem is solved at the beginning of decision window k. Meanwhile, in each time slot during the decision window, DTs estimate the real-time local sensing quality of the sensors. If there is no estimation error, all uplink resources are allocated to local sensory data aggregation, i.e., $o_k = 1$. Otherwise, the estimation error leads to the degradation of the expected local sensing quality that was evaluated at the beginning of the decision window. Therefore, the objective function (23a) can be rewritten in two parts

$$[\mathbf{P}_0]\ \max_{\rho_k, D_k, o_k}\ \underbrace{\sum_{l\in\mathcal{L}}\frac{|\mathcal{N}_l|}{NL}\check{G}_l}_{\text{Part1}} - \underbrace{\frac{1}{N}\sum_{n\in\mathcal{N}}\text{Res}_n\left(\tilde{G}_n, \bar{T}_k\right)}_{\text{Part2}}$$
$$\text{s.t.}\ (23b){-}(23d) \tag{24}$$

where set $\mathcal{N}_l$ represents sensors that cover subarea l, where $\mathcal{N}_l = \{n | l \in \mathcal{L}_n, \forall n\}$. In (24), $\tilde{G}_n = (1/L)\sum_{l\in\mathcal{L}_n}\check{G}_l$ represents the expected final sensing quality can be achieved at sensor n over time, if the estimation is accurate. Part 1 represents the expected final sensing quality analyzed by DTs at the beginning of decision window k, which is equivalent to (23a) if DTs can perfectly estimate the local sensing quality for the sensors. In Part 2, $\text{Res}_n(\tilde{G}_n, \bar{T}_k)$ represents the expected performance degradation resulting from inaccurate estimation by the DT of sensor n, which is related to estimation error $\Delta(\bar{T}_k)$. Such a performance degradation occurs when the ground truth local sensing quality, $\bar{Q}_{n,t}$, is less than the threshold for collaborative sensing while the estimated value, $\hat{Q}_{n,t}$, is not, i.e., the DT of sensor n overestimates the local sensing quality of the sensor. Given threshold ρ_k, the probability that such an overestimation error occurs is

$$P_n^E\left(\bar{T}_k\right) = P\left(\rho_k - \Delta(\bar{T}_k) < \bar{Q}_{n,t} < \rho_k | \Delta(\bar{T}_k) > 0\right). \tag{25}$$

Thus, the function $\text{Res}(\bar{T}_k)$ can be estimated as follows:

$$\text{Res}_n\left(\tilde{G}_n, \bar{T}_k\right) = \int_0^\infty O_{n,t} P\left(\rho_k - \delta < \bar{Q}_{n,t} < \rho_k\right) f_{\Delta^+(\bar{T}_k)}(\delta) d\delta \tag{26}$$

where $O_{n,t} = \tilde{G}_n - \bar{Q}_{n,t}$ represents the loss of final sensing quality when overestimating the local sensing quality. The random variable $\Delta^+(\bar{T}_k)$ represents the estimation error $\Delta(\bar{T}_k)$ under the condition that $\Delta(\bar{T}_k) > 0$.

C. Problem Decomposition

At the beginning of decision window k, the network controller solves $\mathbf{P}_0$ to generate the threshold for collaborative sensing and the resource allocation policy based on the outputs from DTs. We decouple the problem into two subproblems: 1) collaborative sensing optimization and 2) DT management. Specifically, the collaborative sensing optimization problem is defined as follows:

$$[\mathbf{P}_1]\ \max_{\Gamma, \tilde{G}_n, \bar{T}_k}\ \sum_{l\in\mathcal{L}}\frac{|\mathcal{N}_l|}{NL}\check{G}_l \tag{27a}$$

$$\text{s.t.}\ \tilde{G}_n = \frac{1}{L}\sum_{l\in\mathcal{L}_n}\check{G}_l \tag{27b}$$

$$\tilde{G}_n = Z_n, \bar{T}_k = J_{n,k}\quad \forall n$$
$$(23b){-}(23d) \tag{27c}$$

where $\Gamma = \{\rho_k, D_k, o_k\}$ represents the set of optimization variables in $\mathbf{P}_0$. In (27c), Z_n and $J_{n,k}$ are the auxiliary optimization variables that should be equal to the expected final sensing quality of sensor n and the expected update interval $\bar{T}_k$, respectively. These two auxiliary variables are determined by solving the following DT management problem for minimizing performance degradation due to the overestimation error in the DT for each sensor:

$$[\mathbf{P}_2]\ \min_{Z_n, J_{n,k}}\ \text{Res}_n\left(Z_n, J_{n,k}\right) \tag{28a}$$

$$\text{s.t.}\ Z_n = \tilde{G}_n, J_{n,k} = \bar{T}_k\quad \forall n. \tag{28b}$$

Both $\mathbf{P}_1$ and $\mathbf{P}_2$ are nonconvex. Solving $\mathbf{P}_1$ requires predicted CSD models provided by all DTs in a centralized manner at the network controller, and $\mathbf{P}_2$ can be solved by each DT individually based on the estimation error approximated by the DT.

V. Proposed DT-Based Collaborative Sensing Solution

In this section, we present a DT-based collaborative sensing solution to solve $\mathbf{P}_0$. At the beginning of decision window k, DTs predict the CSD models of the corresponding sensors to solve $\mathbf{P}_1$. The algorithm for solving this problem is detailed in Section V-A. Furthermore, each DT approximates

Algorithm 1 Collaborative Sensing Scheme With $o_k = 1$

```
1: Initialize population binary matrix $\mathbb{D}^1$ for $e = 0$.
2: Obtain $\bar{\alpha}_d^{(k)}$ and $\bar{\beta}_d^{(k)}$ from DTs.
3: for episode $e = 1 : E_{\max}$ do
4:     Convert binary population matrix $\mathbb{D}^e$ to the corresponding decimal vector $\mathbf{b}^e = [b_1^e, \dots, b_{N_P}^e]$. Each element represents a candidate downlink bandwidth solution.
5:     for threshold $\rho^d = \rho^1{:}\rho^D$ do
6:         Calculate buffer overflow probability $p_{d,v} = P(\mu_t > \mu_{\max})$ from (7) for all individuals.
7:     end for
8:     Update $\mathbf{b}^e \leftarrow \mathbf{b}^e \backslash \{b_v^e | p_{d,v} > 1 - \varrho, \forall d\}$.
9:     if $\mathbf{b}^e = \emptyset$ then
10:        Return $D_k = 0$.
11:    end if
12:    Update $\boldsymbol{\rho}^e \leftarrow \{\rho^d | p_{d,v} \leq 1 - \varrho, p_{d',v} > 1 - \varrho, \forall d' > d\}$.
13:    for subarea $l \in \mathcal{L}$ do
14:        Calculate the final sensing quality for subarea $l$, i.e., $\check{G}_l^v$, using (20) for all individuals.
15:    end for
16:    Obtain $f_v^e = \sum_l \frac{|\mathcal{N}_l|}{NL} \check{G}_l^v, \forall v$
17:    Find the individual $v^* = \text{argmax}_v f_v^e$.
18:    Selection: Random select $N_P$ individuals from the population with probability $p_v = \frac{f_v^e}{\sum_v (f_v^e)}$, and update binary matrix $\mathbb{D}^{e+1}$.
19:    Crossover: For each individual in $\mathbb{D}^{e+1}$, switch each binary with that of another individual with the probability $p_{\text{crossover}}$, and update $\mathbb{D}^{e+1}$.
20:    Mutation: Flip each binary element in $\mathbb{D}^{e+1}$ with the probability $p_{\text{mutate}}$.
21: end for
22: Output downlink bandwidth $D_k = b_{v^*}^e$ and the corresponding threshold for collaborative sensing $\rho_k = \rho_{v^*}^e$
```

its estimation error. This error model allows for $\mathbf{P}_1$ to be jointly solved with $\mathbf{P}_2$, thereby optimizing the collaborative sensing performance by mitigating the performance degradation due to inaccuracies in the DTs. The joint collaborative sensing and DT management scheme is presented in Section V-B. Finally, we introduce a VAE-based model prediction and DT evaluation method to predict CSD models and approximate the estimation error for each DT in Section V-C.

A. On-Demand Collaborative Sensing

We first solve $\mathbf{P}_1$ when the uplink resource is fully allocated to local sensory data aggregation, i.e., $o_k = 1$. In such a case, DTs are assumed to make accurate estimations on local sensing quality. The optimal downlink bandwidth D_k and the corresponding threshold for collaborative sensing ρ_k are to be determined to maximize the final sensing quality while ensuring the global sensory data delivery delay requirement. A smaller bandwidth allocated for downlink transmissions results in fewer sensors receiving the global sensory data timely, whereas a larger bandwidth for downlink transmissions impedes the local sensory data aggregation.

The proposed resource allocation scheme for on-demand collaborative sensing, where $o_k = 1$ is presented in Algorithm 1. The proposed algorithm is developed from the genetic algorithm. In a genetic algorithm, a population of individuals is created, with each individual representing a candidate resource allocation solution D_k. The population evolves iteratively toward the global optimum, while solving the problem. In iteration episode e, the population is represented by a decimal vector $\mathbf{b}^e$ with length N_P, or the equivalent binary matrix $\mathbb{D}^e$ with a size of $N_P \times N_D$, where N_P represents the number of individuals in a population and N_D is the number of binary bits used to represent a decimal number. The population size N_P should be a large number.

Algorithm 1 consists of three phases. After the population is initialized, the first phase (lines 4–12) aims to find the best threshold for collaborative sensing that satisfies (23b) for each individual of the population. The buffer overflow probability for threshold ρ^d and individual v is denoted by $p_{d,v}$. For each individual, the highest threshold that satisfies (23b) is selected for collaborative sensing. This is because the overall final sensing quality improves as the number of sensors receiving the global sensory data increases. If no individual in the population can satisfy (23b), it indicates that the communication resources are insufficient for collaborative sensing, and the algorithm returns $D_k = 0$. The second phase (lines 13–17) calculates the overall final sensing quality for all individuals and selects the individual that achieves the best final sensing quality. The third phase (lines 18–21) involves population adjustment, where the binary representations of the individuals in the population are modified to improve the quality of the population. Specifically, in line 18, individuals are randomly selected from the current population. The individuals who achieve the higher overall final sensing quality have a higher likelihood of being chosen. Then, the binary representations of selected individuals are combined (crossover in line 19) and randomly mutated (mutation in line 20) to create a new population for the next iteration. Both steps increase population diversity, thereby improving the chance to further improve the sensing performance in the next iteration.

B. Joint Collaborative Sensing and DT Management

Next, we solve the joint collaborative sensing and DT management problem, where $o_k \leq 1$. Specifically, $\mathbf{P}_1$ and $\mathbf{P}_2$ are jointly optimized. Problems $\mathbf{P}_1$ and $\mathbf{P}_2$ are coupled through (27c) and (28b). We first relax $\mathbf{P}_2$ by approximating $\text{Res}_n(Z_n, J_{n,k})$ into a tractable form. When threshold $\rho_k = \rho^d$ is selected for collaborative sensing, the probability that the overestimation occurs, i.e., $\bar{Q}_{n,t} < \rho^d$ and $\hat{Q}_{n,t} > \rho^d$, can be calculated by finding the summation of the probabilities that such an overestimation occurs under the condition $\Delta^+(J_n) > \rho^d - \rho^{d-j} \ \ \forall j \leq d$. In particular, if the ground truth local sensing quality $\bar{Q}_{n,t}$ falls within the interval $[\rho^{d-j}, \rho^{d-j+1}]$, the overestimation occurs when $\Delta^+(J_n) > \rho^d - \rho^{d-j}$. Thus, with threshold ρ^d for collaborative sensing, the overestimation probability in (25) can be estimated as follows:

$$P_n^E\left(J_{n,k};\rho^d\right)=\sum_{j=0}^{d}P_{n,j}^E\left(J_{n,k};\rho^d\right)$$
$$=\sum_{j=1}^{d}P\left(\rho^{d-j}<\bar{Q}_{n,t}<\rho^{d-j+1}\right)P\left(\Delta^+(J_{n,k})>\rho^d-\rho^{d-j}\right). \quad (29)$$

Given threshold ρ^d, $\mathrm{Res}_n(Z_n, J_{n,k})$ can be estimated as

$$\mathrm{Res}_n\left(Z_n,J_{n,k};\rho^d\right)=\sum_{j=1}^{d}\left(Z_n-\mathbb{E}\left[\bar{Q}_{n,t}|\rho^{d-j}<\bar{Q}_{n,t}<\rho^{d-j+1}\right]\right)\times P_{n,j}^E\left(J_{n,k};\rho^d\right). \quad (30)$$

In (30), the expected ground truth local sensing quality under the condition that $\bar{Q}_{n,t}$ falls within the interval $[\rho^{d-j},\rho^{d-j+1}]$ can be estimated as

$$\mathbb{E}\left[\bar{Q}_{n,t}|\rho^{d-j}<\bar{Q}_{n,t}<\rho^{d-j+1}\right]=\rho^{d-j}+\frac{1}{2}\left(\rho^{d-j}+\rho^{d-j+1}\right). \quad (31)$$

In addition, the probability $P(\rho^{d-j}<\bar{Q}_{n,t}<\rho^{d-j+1})$ can be estimated by the predicted CSD models from DT n as follows:

$$P\left(\rho^{d-j}<\bar{Q}_{n,t}<\rho^{d-j+1}\right)=\frac{\hat{\alpha}_{n,\rho^{d-j+1}}}{\hat{\alpha}_{n,\rho^{d-j+1}}+\hat{\beta}_{n,\rho^{d-j+1}}}-\frac{\hat{\alpha}_{n,\rho^{j}}}{\hat{\alpha}_{n,\rho^{d-j}}+\hat{\beta}_{n,\rho^{d-j}}}. \quad (32)$$

In this way, $\mathrm{Res}_n(Z_n, J_{n,k})$ can be derived based on the information provided by DTs given threshold ρ^d for collaborative sensing.

We then solve $\mathbf{P}_1$ and $\mathbf{P}_2$ iteratively using the alternating direction method of multipliers (ADMMs) method [31] for decoupling the problem and the relaxed heavy ball ADMM (R-HB-ADMM) method [32] to accelerate convergence. Let $\mathbf{W}$ denote the vector that combines $[\tilde{G}_1,\ldots,\tilde{G}_N]$ and $[\bar{T}_k,\ldots,\bar{T}_k]$ with the length of $2N$. Let $\mathbf{V}$ denote the vector that combines $[Z_1,\ldots,Z_N]$ and $[J_{1,k},\ldots,J_{N,k}]$. Let $\mathbf{W}_n$ and $\mathbf{V}_n$ represent vectors $[\tilde{G}_n,\bar{T}_k]$ and $[Z_n,J_{n,k}]$, respectively. Using the R-HB-ADMM methods, in iteration m, the optimization variables in $\mathbf{P}_2$ are updated using the augmented Lagrangian method by solving

$$\min_{\mathbf{V}_n}\ F_{2,n}=\mathrm{Res}_n\left(Z_n,J_{n,k};\rho^d\right)+\frac{\epsilon}{2}\|\mathbf{V}_n-\check{\mathbf{W}}^m+\check{u}^m\|_2^2. \quad (33)$$

Similarly, the optimization variables in $\mathbf{P}_1$ are updated in iteration m by solving

$$\max_{\Gamma,\mathbf{W}}\quad F_1=\sum_{l\in\mathcal{L}}\frac{|\mathcal{N}_l|}{NL}\check{G}_l-\frac{\epsilon}{2}\|\omega_1\mathbf{V}^{m+1}+(1-\omega_1)\check{\mathbf{W}}^m-\mathbf{W}+\check{u}^m\|_2^2$$
$$\text{s.t.}\quad (23b)-(23d). \quad (34)$$

In (34) and (33), the auxiliary variables $\check{\mathbf{W}}^m$ and $\check{u}^m$ in R-HB-ADMM are updated using the following rules:

$$u^{m+1}\leftarrow\check{u}^m+\omega_1\mathbf{V}^{m+1}+(1-\omega_1)\check{\mathbf{W}}^m-\mathbf{W}^{m+1} \quad (35a)$$
$$\check{u}^{m+1}\leftarrow u^{m+1}+\omega_2\left(u^{m+1}-u^m\right) \quad (35b)$$
$$\check{\mathbf{W}}^{m+1}\leftarrow\mathbf{W}^{m+1}+\omega_2\left(\check{\mathbf{W}}^{m+1}-\check{\mathbf{W}}^m\right). \quad (35c)$$

Algorithm 2 Collaborative Sensing Scheme With $o_k \leq 1$

```
1:  Initialize $u^0$, $\check{\mathbf{W}}^0$, $\mathbf{W}^0$, and $\mathbf{V}^0$.
2:  Obtain $\bar{\alpha}_d^{(k)}$ and $\bar{\beta}_d^{(k)}$ from DTs
3:  for iteration $m = 1 : M_{\max}$ do
4:      for sensor $n = 1 : N$ do
5:          Update the auxiliary variables in $m$th iteration using (35) to obtain $\check{\mathbf{W}}^m(\rho^d)$ and $\check{u}^m(\rho^d)$, $\forall\rho^d \in \boldsymbol{\rho}^e$.
6:          Find $Z_n^{m+1}(\rho^d)$, $\forall\rho^d \in \boldsymbol{\rho}^e$ using (36),.
7:          Apply the genetic algorithm to solve (33) and obtain $\mathbf{V}_n^{m+1}(\rho^d)$, $\forall\rho^d \in \boldsymbol{\rho}^e$.
8:      end for
9:      Initialize binary population matrices $\mathbb{D}^1$ and $\mathbb{O}^1$ with size $N_P \times N_D$, respectively.
10:     for episode $e = 1$: $E_{\max}$ do
11:         Convert matrices $\mathbb{D}^e$ and $\mathbb{O}^e$ to corresponding decimal vectors $\mathbf{b}^e$ and $\mathbf{o}^e$, respectively.
12:         Find threshold for collaborative sensing $\boldsymbol{\rho}^e$ using Lines 5-12 of Algorithm 1.
13:         Obtain objective function value $F_{1,v}^e$ for all indiviuals of the population using $\mathbf{V}_n^{m+1}(\boldsymbol{\rho}^e)$.
14:         Find the individual $v^* = \mathrm{argmax}_v F_{1,v}^e$.
15:         Diversify the population in $\mathbb{D}^e$ and $\mathbb{O}^e$ using Lines 18-20 of Algorithm 1.
16:     end for
17:     Update $\mathbf{W}^{m+1}(\boldsymbol{\rho}^e)$ based on resource allocation policy $D_k = b_{v^*}^e$ and $o_k = o_{v^*}^e$.
18: end for
19: Output optimal resource allocation policy $D_k$ and $o_k$ and the threshold for collaborative sensing $\rho_k = \rho_{v^*}^e$.
```

In (34), (33), and (35), ϵ, ω_1, and ω_2 are the parameters in the R-HB-ADMM method.

Inspired by R-HB-ADMM, we develop a joint DT management and collaborative sensing scheme, as presented in Algorithm 2. This algorithm aims to balance the expected final sensing quality and performance degradation resulting from DT estimation error. First, for the DT of each sensor, lines 6–8 compute the solutions to (33) for all potential thresholds for collaborative sensing. The optimal Z_n in each iteration can be obtained directly by solving $\nabla_{Z_n}F_{2,n}=0$, where Z_n^{m+1} given threshold ρ^d can be obtained as follows:

$$Z_n^{m+1}\left(\rho^d\right)\leftarrow\tilde{G}_n^m-u_h^m-\sum_{j=1}^{d}P_{n,j}^E\left(J_{n,k}^m;\rho^d\right)/\epsilon. \quad (36)$$

The optimal $J_{n,k}^m$, given ρ^d, is obtained using a genetic algorithm. After solving (33), lines 11 to 18 solves (34). This part of Algorithm 2 generates and evolves two populations, one for finding D_k and another for finding o_k. Similar to Algorithm 1, the optimal threshold for collaborative sensing is determined for each individual in line 14. The optimization variables in the two problems, i.e., (33) and (34), are updated iteratively till the maximum iteration number is reached, i.e., $M_{\max}$. Note that, Algorithm 2 is executed at the beginning of every decision window. The output provides the policy for real-time collaborative sensing over time slots in the subsequent

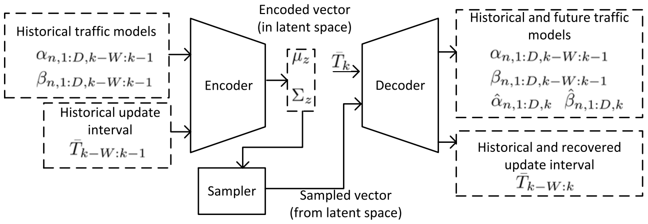

Historical traffic models
$\alpha_{n,1:D,k-W:k-1}$
$\beta_{n,1:D,k-W:k-1}$
Historical update interval
$\bar{T}_{k-W:k-1}$
Encoder
Encoded vector (in latent space)
μ_z
Σ_z
Sampler
Sampled vector (from latent space)
$\bar{T}_k$
Decoder
Historical and future traffic models
$\alpha_{n,1:D,k-W:k-1}$
$\beta_{n,1:D,k-W:k-1}$
$\hat{\alpha}_{n,1:D,k}$ $\hat{\beta}_{n,1:D,k}$
Historical and recovered update interval
$\bar{T}_{k-W:k}$

Fig. 6. Structure of the proposed VAE.

decision window. The processing time of Algorithm 2 can be considered negligible since it operates once in each decision window which contains multiple time slots.

C. VAE-Based Model Prediction and DT Evaluation

The solutions provided in Algorithms 1 and 2 are based on the predicted future CSD models ($\hat{\alpha}^{(k)}_{n,d}$, $\hat{\beta}^{(k)}_{n,d}$) and the known estimation error model $\Delta(\bar{T}_k)$, which are generated by DTs. In this section, we introduce a VAE to generate the above information based on the historical CSD models and the average update intervals in preceding W decision windows. Compared to other prediction methods like recurrent neural networks, the VAE allows DTs to learn the intrinsic dynamics of environmental changes from noisy estimated local sensing quality data. Additionally, DTs can assess their estimation error corresponding to different update intervals using the proposed VAE.

The proposed VAE structure is illustrated in Fig. 6. Each DT deploys a VAE for model prediction and DT evaluation. The VAE takes the historical CSD models and update intervals of the corresponding sensor in the preceding W decision windows as the input, denoted by vector $\mathbf{x}$. The proposed VAE consists of an encoder and a decoder. The encoder is a neural network trained to learn a latent variable space $\mathcal{Z}$ as the data feature from the input, in which the latent variable space can be represented by a probability density function $P_\theta(\mathbf{z})$ for a latent variable $\mathbf{z} \in \mathcal{Z}$. In [33], the latent variable $\mathbf{z}$ is modeled as a normal distribution denoted by $\hat{R}(\mathbf{z}|\mathbf{x})$. Thus, the encoder learns the encoded vector, i.e., the mean and standard deviation of $\hat{R}(\mathbf{z}|\mathbf{x})$, denoted by μ_z and Σ_z, to represent $P_\theta(\mathbf{z})$. In addition, the decoder is another neural network that reconstructs the input from the distribution $\hat{R}(\mathbf{z}|\mathbf{x})$. With this approach, the encoder can capture the randomness of estimation errors in historical demand models and explore the true dynamics of resource demand for collaborative sensing.

In addition, we introduce a prediction function within the decoder. This function predicts the CSD models in time slot k based on a given expected update interval $\bar{T}_k$, using information from $\hat{R}(\mathbf{z}|\mathbf{x})$. If the VAE is well-trained, when the input $\bar{T}_k$ is zero, the decoder will generate accurate future CSD models for resource allocations. Furthermore, the potential expected update intervals $\bar{T}_k$ can be fed to the decoder to emulate the CSD models with estimation errors caused by the corresponding update intervals, allowing the DT to approximate the estimation error given the expected interval $\bar{T}_k$.

The proposed VAE-based model prediction and DT evaluation involves three parts: 1) VAE inference; 2) estimation error evaluation; and 3) VAE retraining. First, in VAE inference, at the beginning of decision window k, DT n retrieves historical CSD models from decision windows $k-W$ to $k-1$, along with the corresponding historical update intervals. The encoder generates encoded vector (μ_z, Σ_z) to determine a normal distribution that allows a sampler to draw a sampled vector $\mathbf{z}$ from this distribution. The sampled vector and candidate $\bar{T}_k$ serve as inputs to the decoder, which produces the recovered and future CSD models as well as the recovered update interval. When predicting the future demand models for making collaborative sensing and resource allocation decisions, we set $\bar{T}_k$ to zero as the input of the decoder to generate the CSD models without the estimation error.

Second, in estimation error evaluation, each DT uses its VAE to approximate the estimation error by emulating CSD models with $\bar{T}_k > 0$. This is achieved by sampling multiple $\bar{T}_k$ values as the possible update intervals and using them as input for the decoder. Since we consider an unbiased estimation in which the mean of $\Delta(\bar{T}_k)$ is 0, we can have the following relation on the CSD models generated with input $\bar{T}_k = y$ and $\bar{T}_k = 0$ fed to the decoder:

$$P\Big(\tilde{G}_n + \Delta(y) < \rho^d|\bar{T}_k = y\Big) \;= P\Big(\tilde{G}_n < \rho^d|\bar{T}_k = 0\Big). \quad (37)$$

The left-hand side of the equation can be rewritten as follows:

$$\begin{aligned} &P\Big(\tilde{G}_n < \rho^d - \Delta(y)|\bar{T}_k = y\Big) \\ &\quad = \int_\delta P\Big(\tilde{G}_n < \rho^d - \delta|\bar{T}_k = y\Big) f_{\Delta(y)}(\delta) d\delta. \end{aligned}$$

Given the small gap between the thresholds in set $[\rho^1, \ldots, \rho^D]$, we can replace $\rho^d - \delta$ by an element from $[\rho^1, \ldots, \rho^D]$, denoted by ρ^x. This allows us to estimate (37) as follows:

$$\begin{aligned} &\sum_{x=1}^{D} P\Big(\tilde{G}_n < \rho^x|\bar{T}_k = y\Big)\mathbb{Q}\Big(\Delta(y), \rho^d, \rho^x\Big) \\ &\quad = P\Big(\tilde{G}_n < \rho^d|\bar{T}_k = 0\Big). \end{aligned} \quad (38)$$

In (38), $\mathbb{Q}(\Delta(y), \rho^d, \rho^x)$ represents the probability of the estimation error falling within the interval (ρ^d, ρ^x) when $\rho^d < \rho^x$ or (ρ^x, ρ^d) when $\rho^x < \rho^d$. This probability can be calculated by

$$\mathbb{Q}\Big(\Delta(y), \rho^d, \rho^x\Big) = \left|Q\left(\frac{\rho^d}{\sigma_n(y)}\right) - Q\left(\frac{\rho^x}{\sigma_n(y)}\right)\right|$$

where $Q(\cdot)$ is the Q-function in statistics. The conditional probability in (38) can be obtained from the proposed VAE with different $\bar{T}_k$ fed to the decoder

$$P\Big(\tilde{G}_n < \rho^x|\bar{T}_k = y\Big) \;= \frac{\hat{\alpha}_{n,x}\big(k|\bar{T}_k = y\big)}{\hat{\alpha}_{n,x}\big(k|\bar{T}_k = y\big) + \hat{\beta}_{n,x}\big(k|\bar{T}_k = y\big)}.$$

The ideal parameters $\mathcal{K}_n$ in the parameterized function $\sigma_n(\bar{T}_k)$ should ensure that the left-hand of (38) equals to right-hand sides. Thus, the parameters $\mathcal{K}_n$ can be obtained through the information provided by the VAE as follows[4]:

$$\mathcal{K}_n^* = \text{argmin}_{\mathcal{K}_n} \sum_{d=1}^{D} \left\{ \sum_{x=1}^{D} P\left(\tilde{G}_n < \rho^x | \bar{T}_k = y\right) \mathbb{Q}\left(\Delta(y), \rho^d, \rho^x\right) - P\left(\tilde{G}_n < \rho^d | \bar{T}_k = 0\right) \right\}^2. \tag{39}$$

Finally, in VAE retraining, the neural networks in the VAE model are retrained at the end of each decision window. Each DT fetches historical CSD models from decision windows $k - W$ to k and inputs them into the encoder and decoder. The loss function for retraining the VAE is defined as follows [33]:

$$\mathcal{L} = \sum_{w} \left\{ KL\left[\hat{R}_\lambda(\mathbf{z}|\mathbf{x}_w) || P_\theta(\mathbf{z})\right] - \mathbb{E}_{\boldsymbol{\epsilon} \sim \mathcal{N}(0,I)}\left[P_\phi(\mathbf{x}_w | \mathbf{z}(\boldsymbol{\epsilon}, \mathbf{x}_w))\right] \right\} \tag{40}$$

where λ and ϕ represent the weights of the neural networks in the encoder and decoder, respectively. Additionally, $\boldsymbol{\epsilon}$ is an auxiliary noise variable sampled from a standard normal distribution. In the equation, $\mathbf{z}(\boldsymbol{\epsilon}, \mathbf{x}_i)$ represents the sampled value of $\mathbf{z}$ from the encoded vector μ_z and Σ_z for input data point $\mathbf{x}_i$, which can be obtained from $\mathbf{z}(\boldsymbol{\epsilon}, \mathbf{x}_i) = \mu_z(\mathbf{x}_i) + \Sigma_z(\mathbf{x}_i)\boldsymbol{\epsilon}$.

The proposed VAE-based data prediction and DT evaluation play a critical role in establishing a closed-loop resource allocation in collaborative sensing. The VAE forecasts future resource demands using historical data to support proactive resource allocation. Additionally, the VAE analyzes the system performance, i.e., historical CSD models, on the historical resource allocation decision, i.e., update interval T_k, and evaluates the estimation error resulting from DTs. This evaluation enables the network controller to optimize collaborative sensing policies through the DTs while taking into account DT estimation error and maintenance costs.

VI. Simulation Results

We consider a 100 m × 100 m workspace with $N = 50$ sensors. Each sensor is capable of sensing its surroundings within a radius of 10 m, and its sensing coverage may be occasionally blocked by moving objects traversing the workspace. The number of moving objects accessing the workspace among decision windows is time-varying, where the pattern is generated by an autoregressive integrated moving average (ARIMA) model. The speeds of the moving objects are uniformly selected from [1, 4] m/s. The duration of a time slot τ is 1 s, and a decision window contains 120 slots. Within each time slot, the number of moving objects is constant. The size of local sensory data V and the quality status V_L are 2 and 0.3 MB, respectively. For both uplink and downlink transmissions, the noise power is set at -160 dBm/Hz, and the received signal power S_U and S_D are 50 dBm. The constraint on the downlink buffer overflow probability $1 - \varrho$ is 0.1. In Algorithm 1, the genetic algorithm operates with the following parameters: $N_P = 100$, $E_{\max} = 10$, $p_{\text{crossover}} = 0.8$, and $p_{\text{mutate}} = 0.003$. For R-HB-ADMM in Algorithm 2, the parameters are set as follows: $\epsilon = 1$, $\omega_1 = 0.8$, and $\omega_2 = 0.1$.

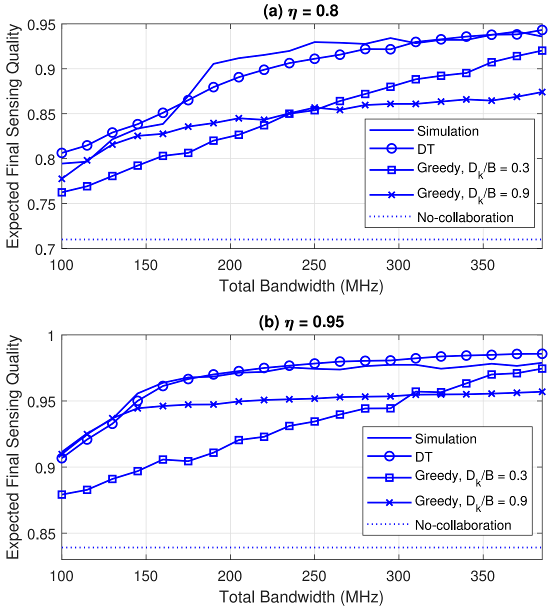


Fig. 7. Expected final sensing quality per sensor versus overall bandwidth B.

A. DT-Based Collaborative Sensing and Resource Allocation

In this section, we demonstrate the performance of the proposed collaborative sensing scheme achieved by DTs. The sensing performance, i.e., the expected final sensing quality per sensor within a decision window, under the condition that $o_k = 1$ is shown in Fig. 7. We first compare the analytical performance using the models in DTs, denoted by "DT" in the figure, with the simulated collaborative sensing performance. Two degradation parameters are evaluated, i.e., $\eta = 0.95$ and $\eta = 0.8$. We observe that the difference between the analytical and the simulated results is very small, which validates that our proposed DT framework can accurately characterize the resource demand in collaborative sensing. In addition, we compare the performance of our proposed DT-based scheme with two benchmark schemes as follows.

1) *Greedy:* The AP delivers the global sensory data to all sensors, i.e., $\rho_k = \rho^D = 1$, with a fixed amount of communication resources allocated to the downlink transmissions.
2) *No-Collaboration:* No sensor receives the global sensory data from the AP, i.e., $\rho_k = \rho^1 = 0$.

Compared to the no-collaboration scheme, the proposed collaborative sensing scheme can significantly enhance sensing performance through sensory data sharing. Furthermore, compared to the greedy scheme, our proposed scheme improves the final sensing quality by optimizing resource allocation and thresholds for collaborative sensing. In the greedy scheme, when $D_k/B = 0.9$, less bandwidth resource is allocated to

[4] We use a linear model to approximate the function $\sigma_n(\bar{T}_k)$, where $\sigma_n(\bar{T}_k) = K_n \times \bar{T}_k$, and the parameter $K_n > 0$. However, the proposed model evaluation method is not limited to a linear model.

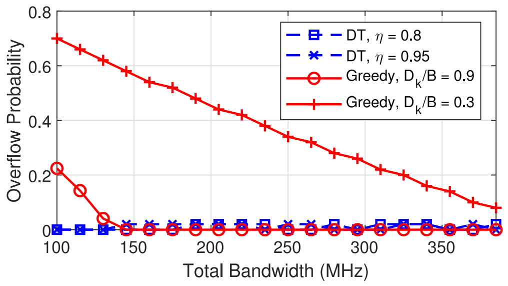


Fig. 8. Buffer overflow probability versus overall bandwidth B.

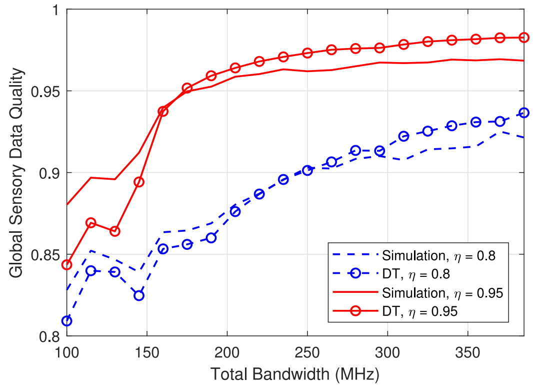


Fig. 9. Global sensing quality versus overall bandwidth B.

sensing data aggregation. The performance gap between the DT and greedy schemes increases as overall bandwidth B increases due to the low global sensing quality, especially when η is low. Meanwhile, when $D_k/B = 0.3$, more bandwidth resource is allocated to sensing data aggregation. However, it may lead to low final sensing quality due to insufficient downlink bandwidth for timely delivering the global sensory data.

The downlink buffer overflow probability for both the proposed DT scheme and the greedy scheme is shown in Fig. 8. Although the final sensing quality in the Greedy case is comparable to that of the proposed scheme when overall bandwidth B is small, the greedy case does not consistently maintain the overflow probability under the required limit $1 - \varrho$. In contrast, the proposed scheme effectively limits the overflow probability, simultaneously maximizing the overall sensing performance.

The comparison between the analytical performance provided by DTs and the simulated performance on the global sensing quality is shown in Fig. 9. The simulation results demonstrate the accuracy of the analytical performance evaluated by the mathematical models in DTs. Furthermore, when the overall bandwidth is below 200 MHz, the quality of the sensory data exhibits fluctuations. This is because the threshold for collaborative sensing is selected from a discrete $\boldsymbol{\rho}$. The results of the selected threshold for collaborative sensing ρ_k and resources allocated to the downlink transmissions D_k are shown in Fig. 10. Below the 200 MHz bandwidth, the selected threshold increases as the overall bandwidth increases. As the optimal threshold may reside within the interval between two adjacent thresholds in $\boldsymbol{\rho}$, the corresponding resource allocation is subject to fluctuations. As the difference between two adjacent thresholds in $\boldsymbol{\rho}$ decreases, the quality of global sensory data is expected to increase steadily with overall bandwidth since sufficient resources are allocated for data aggregation without violating the constraints on the global sensory data delivery.

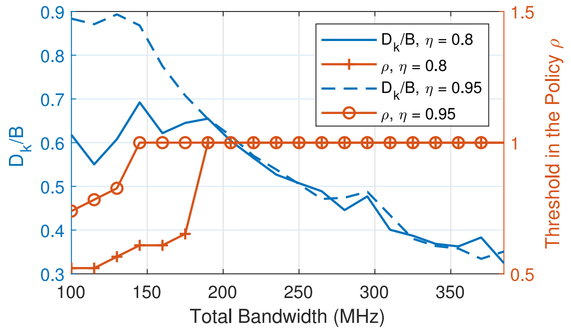
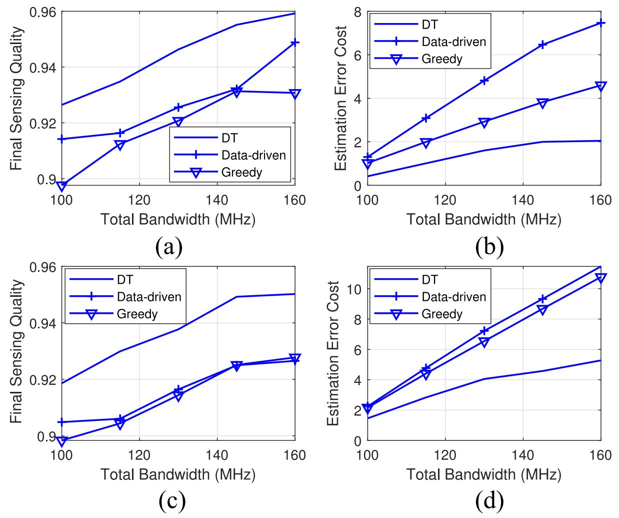


Fig. 10. Resource allocation results and the selected threshold for collaborative sensing versus overall bandwidth B.



(a) (b) (c) (d)

Fig. 11. (a) Final sensing quality versus overall bandwidth B with the DT estimation error when $K = 0.1$. (b) Overall estimation error costs when $K = 0.1$. (c) Final sensing quality versus overall bandwidth B with the DT estimation error when $K = 0.3$. (d) Overall estimation error costs when $K = 0.3$.

We next demonstrate the resource allocation results for DTs with estimation errors. In this scenario, $o_k \leq 1$. We utilize a linear model for the standard deviation of the estimation error distribution, where $\sigma_n(\bar{T}_k) = K\bar{T}_k$ and $\mathcal{K}_n = K \quad \forall n$. The sensing performance when $K = 0.1$ and $K = 0.3$ is shown in Fig. 11(a) and (c), respectively. We compare the performance with a data-driven resource management approach, in which the AP directly uses the data set in the DTs for resource allocation without evaluating the data accuracy. The performance degradation resulting from inaccurate status estimation, i.e., $\sum_n \mathrm{Res}_n(\tilde{G}_n, \bar{T}_k)$, is shown in Fig. 11(b) and (d). As shown in Fig. 11, in comparison with the greedy scheme and the data-driven approach, our proposed scheme achieves the highest final sensing quality with a low estimation

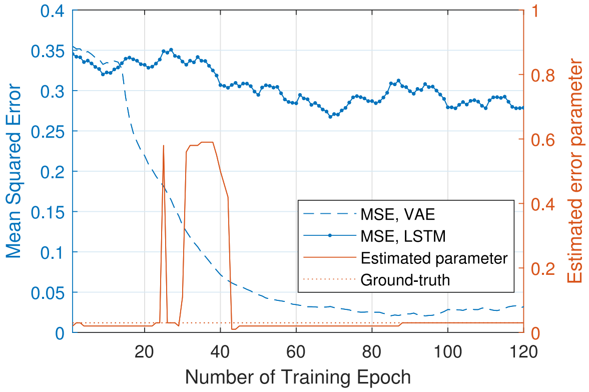


Fig. 12. MSE between the predicted CSD model parameters and the derived estimated error parameters across training epochs.

error. This is because our proposed scheme can fine tune the resource allocation policy for joint resource and DT management while taking into account the estimation accuracy of DTs. A sufficient amount of resource is allocated to collect additional data for DTs to compensate for their inaccuracies.

B. *VAE-Based Data Prediction and Model Evaluation*

In this section, we demonstrate the efficacy of our proposed VAE-based model prediction and DT evaluation method. We consider a time-varying number of moving objects across decision windows. The VAE examines the historical data in the past 50 decision windows to capture the dynamics and forecast future CSD models with a learning rate of 0.0002. The encoder structures in the VAE includes an LSTM layer with 20 units, followed by four fully connected layers with [518, 256, 128, 20] neurons. The decoder structures in the VAE includes three fully connected layers with [128, 256, 518] neurons, followed by an LSTM layer with 20 units. A linear model on the standard deviation of the estimation error is adopted with $\sigma_n(\bar{T}_k) = K\bar{T}_k$ and $K = 0.03$. The VAE undergoes retraining once every five decision windows, referred to as a training epoch.

We compare the proposed VAE-based method with an LSTM method in terms of model prediction accuracy, and the comparison results are shown in Fig. 12. The mean-squared error (MSE) between the VAE predicted and the actual CSD models in a decision window decreases significantly as the index of training epochs increases. On the contrary, the LSTM-based method struggles to learn the dynamics from the historical CSD models with the estimation error. Moreover, the efficiency of our proposed estimation error evaluation method is shown in the figure. After 90 training epochs, the DT successfully learns the parameter $K = 0.03$ by analyzing the predicted CSD models with varying candidate time slot length $\bar{T}_k$ as the input of the decoder.

VII. Conclusion

In this article, we have developed a novel resource allocation solution for collaborative sensing in an IIoT scenario. Our proposed hybrid data- and model-driven resource allocation framework, empowered by DTs, facilitates on-demand resource allocation by efficiently characterizing sensor-level resource demand into resource allocation policies. Within this framework, we have developed a joint collaborative sensing and resource allocation scheme to coordinate resources allocated for enhancing sensing performance and supporting DT operations. In addition, we have created a VAE-based data prediction and DT evaluation model, which enables a closed-loop collaborative sensing approach that evaluates the model accuracy of DTs. The proposed DT-based solution sets an example for on-demand resource allocation to efficiently support the extensive intelligent connections expected in next-generation networks. Our future research will delve into leveraging DTs to enhance reliability in communication and sensing and enable secure interactions between humans and machines in Industry 5.0.

Appendix
Proof of (20)

We prove (20) based on the following four cases.

A. *Case 1*

No global sensory data is delivered to the sensor. With the probability of κ, the sensor can sense subarea l successfully, and the maximum AoI of the local sensory data is τ. The probability of such a case occurring is $P^G_{l,1} = \kappa P(\hat{Q}_{n,t} \geq \rho_d)$, which can be further simplified in (22a).

B. *Case 2*

The sensor receives the global sensory data for subarea l, and the global sensory data is from the local sensory data captured by other sensors within a time slot. In such a case, $I_l \leq \hat{I}$. The corresponding probability on $I_l \leq \hat{I}$ is $P(I_l \leq \hat{I}) = 1 - q_l^{\hat{I}+1}$, where $q_l = 1 - g_l$. Accordingly, the probability that case 2 occurs can be obtained as (22a). In addition, since I_l follows a geometric distribution, the PMF of I_l when $I_l \leq \hat{I}$ can be formulated as:

$$P\left(I_l = x | I_l \leq \hat{I}\right) = \frac{P\left(I_l = x \cup I_l \leq \hat{I}\right)}{P\left(I_l \leq \hat{I}\right)} = \frac{q_l^{x-1} g_l}{1 - q_l^{\hat{I}+1}}.$$

To determine the expected AoI of the received global sensory data for subarea l as (19), we calculate the mean and the second moment of I_l given that $I_l \leq \hat{I}$. The mean is derived as follows:

$$\begin{aligned}\mathbf{E}\left[I_l | I_l \leq \hat{I}\right] &= \frac{\sum_{i=0}^{\hat{I}} i q_l^i g_l}{1 - q_l^{\hat{I}+1}} = \frac{g_l q_l}{1 - q_l^{\hat{I}+1}} \frac{d}{dq} \sum_{i=1}^{\hat{I}} q_l^i \\ &= \frac{q_l\left(1 - q_l^{\hat{I}} - \hat{I} q_l^{\hat{I}-1}\right)}{g_l\left(1 - q_l^{\hat{I}+1}\right)}. \qquad (41)\end{aligned}$$

The second moment of I_l with $I_l \leq \hat{I}$ is

$$\mathbf{E}\left[I_l^2 | I_l \leq \hat{I}\right] = \frac{\sum_{i=0}^{\hat{I}} i2 q_l^i g_l}{1 - q_l^{\hat{I}+1}} = \mathbf{E}\left[I_l | I_l \leq \hat{I}\right] + \frac{(T_A + T_B)}{g_l\left(1 - q_l^{\hat{I}+1}\right)}$$

where

$$T_A = \frac{q_l}{g_l}\left[2q_l + \left(\hat{I}-1\right)\left(\hat{I}+2\right) q_l^{\left(\hat{I}+1\right)} - \hat{I}\left(\hat{I}+1\right) q_l^{\hat{I}})\right] \tag{42a}$$

$$T_B = \frac{q_l}{g_l^2}\left[2\left(1-q_l^{\hat{I}}\right)q_l^2 - \hat{I}g_l q_l^{\hat{I}+1}\right]. \tag{42b}$$

Using (19), the expected AoI of the global sensory data for subarea l received at a sensor is determined by (21a).

C. Case 3

The sensor cannot successfully sense subarea l and receives the global sensory data from the AP with $I_l > \hat{I}$. In this case, the global sensory data is used at the sensor as the final sensory data. Consequently, the probability of such a case can be formulated as

$$P_{l,3}^G = (1-\kappa)P\left(\hat{Q}_{n,t} < \rho_d\right)P\left(I_l > \hat{I}\right). \tag{43}$$

To obtain the closed-form expected AoI of the received global sensory data, we can derive the mean of I_l under the condition $I_l > \hat{I}$ as

$$\mathbf{E}\left[I_l | I_l > \hat{I}\right] = \frac{q_l\left(q_l^{\hat{I}} + \hat{I}q_l^{\hat{I}-1}\right)}{g_l\left(q_l^{\hat{I}+1}\right)} \tag{44}$$

and the corresponding second moment is

$$\mathbf{E}\left[I_l^2 | I_l > \hat{I}\right] = \frac{2-g_l}{q_l^{\hat{I}+1}g_l^2} - \frac{\mathbf{E}\left[I_l^2 | I_l < \hat{I}\right]\left(1-q_l^{\hat{I}+1}\right)}{q_l^{\hat{I}+1}}. \tag{45}$$

The derivation process is similar to Case 2 and omitted.

D. Case 4

The sensor can successfully sense subarea l and receive the global sensory data from the AP with $I_l > \hat{I}$. The received global sensory data for subarea l is outdated compared to the local one. In this case, the sensor uses the local sensory data for subarea l as the final one, and the probability that such a case occurs is (22b).

References

[1] H. R. Chi, C. K. Wu, N.-F. Huang, K.-F. Tsang, and A. Radwan, "A survey of network automation for industrial Internet-of-Things toward Industry 5.0," *IEEE Trans. Ind. Informat.*, vol. 19, no. 2, pp. 2065–2077, Feb. 2023.

[2] W. Xian, K. Yu, F. Han, L. Fang, D. He, and Q.-L. Han, "Advanced manufacturing in industry 5.0: A survey of key enabling technologies and future trends," *IEEE Trans. Ind. Informat.*, vol. 20, no. 2, pp. 1055–1068, Feb. 2024.

[3] P. K. R. Maddikunta et al., "Industry 5.0: A survey on enabling technologies and potential applications," *J. Ind. Inf. Integr.*, vol. 26, Mar. 2022, Art. no. 100257.

[4] S. Kianoush, S. Savazzi, M. Beschi, S. Sigg, and V. Rampa, "A multisensory edge-cloud platform for opportunistic radio sensing in cobot environments," *IEEE Internet Things J.*, vol. 8, no. 2, pp. 1154–1168, Jan. 2021.

[5] S. He, K. Shi, C. Liu, B. Guo, J. Chen, and Z. Shi, "Collaborative sensing in Internet of Things: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 3, pp. 1435–1474, 3rd Quart., 2022.

[6] M. K. Abdel-Aziz, C. Perfecto, S. Samarakoon, M. Bennis, and W. Saad, "Vehicular cooperative perception through action branching and federated reinforcement learning," *IEEE Trans. Commun.*, vol. 70, no. 2, pp. 891–903, Feb. 2022.

[7] G. Luo et al., "EdgeCooper: Network-aware cooperative LiDAR perception for enhanced vehicular awareness," *IEEE J. Sel. Areas Commun.*, vol. 42, no. 1, pp. 207–222, Jan. 2024.

[8] X. Deng, Z. Tang, L. T. Yang, M. Lin, and B. Wang, "Confident information coverage hole healing in hybrid industrial wireless sensor networks," *IEEE Trans. Ind. Informat.*, vol. 14, no. 5, pp. 2220–2229, May 2018.

[9] S. Huang, M. Gao, L. Liu, J. Chen, and J. Zhang, "Collision detection for Cobots: A back-input compensation approach," *IEEE/ASME Trans. Mechatronic*, vol. 27, no. 6, pp. 4951–4962, Dec. 2022.

[10] F. Ghassemi and V. Krishnamurthy, "Separable approximation for solving the sensor subset selection problem," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 47, no. 1, pp. 557–568, Jan. 2011.

[11] N. Sadeghzadeh-Nokhodberiz and J. Poshtan, "Distributed interacting multiple filters for fault diagnosis of navigation sensors in a robotic system," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 7, pp. 1383–1393, Jul. 2017.

[12] H. Wu, Z. Zhang, C. Jiao, C. Li, and T. Q. S. Quek, "Learn to sense: A Meta-learning-based sensing and fusion framework for wireless sensor networks," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8215–8227, Oct. 2019.

[13] W. Zhang et al., "Optimizing federated learning in distributed industrial IoT: A multi-agent approach," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 12, pp. 3688–3703, Dec. 2021.

[14] X. Shen, J. Gao, W. Wu, M. Li, C. Zhou, and W. Zhuang, "Holistic network virtualization and pervasive network intelligence for 6G," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 1, pp. 1–30, 1st Quart., 2022.

[15] E. VanDerHorn and S. Mahadevan, "Digital twin: Generalization, characterization and implementation," *Decis. Support Syst.*, vol. 145, Jun. 2021, Art. no. 113524.

[16] J. Yang, X. Wu, and J. Wu, "Optimal scheduling of collaborative sensing in energy harvesting sensor networks," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 3, pp. 512–523, Mar. 2015.

[17] D. Ma, G. Lan, M. Hassan, W. Hu, and S. K. Das, "Sensing, computing, and communications for energy harvesting IoTs: A survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 1222–1250, 2nd Quart., 2020.

[18] R. Liu et al., "ROS-based collaborative driving framework in autonomous vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 72, no. 6, pp. 6987–6999, Jun. 2023.

[19] W. Jiang, B. Ai, C. Shen, M. Li, and X. Shen, "Age-of-information minimization for UAV-based multi-view sensing and communication," *IEEE Trans. Veh. Technol.*, vol. 73, no. 1, pp. 1100–1114, Jan. 2024.

[20] H. Li, M. Tsukada, F. Nashashibi, and M. Parent, "Multivehicle cooperative local mapping: A methodology based on occupancy grid map merging," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 5, pp. 2089–2100, Oct. 2014.

[21] L. Xie, S. Song, Y. C. Eldar, and K. B. Letaief, "Collaborative sensing in perceptive mobile networks: Opportunities and challenges," *IEEE Wireless Commun.*, vol. 30, no. 1, pp. 16–23, Feb. 2023.

[22] X. Zhang, M. Peng, S. Yan, and Y. Sun, "Joint communication and computation resource allocation in fog-based vehicular networks," *IEEE Internet Things J.*, vol. 9, no. 15, pp. 13195–13208, Aug. 2022.

[23] Q. Xie, X. Zhou, T. Qiu, Q. Zhang, and W. Qu, "Soft actor–critic-based multilevel cooperative perception for connected autonomous vehicles," *IEEE Internet Things J.*, vol. 9, no. 21, pp. 21370–21381, Nov. 2022.

[24] 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.

[25] R. Ding, F. Zhou, Q. Wu, C. Dong, Z. Han, and O. A. Dobre, "Data and knowledge dual-driven automatic modulation classification for 6G wireless communications," *IEEE Trans. Wireless Commun.*, vol. 23, no. 5, pp. 4228–4242, May 2024.

[26] C. Zhou, J. Gao, M. Li, X. Shen, and W. Zhuang, "Digital twin-empowered network planning for multi-tier computing," *J. Commun. Inf. Netw.*, vol. 7, no. 3, pp. 221–238, Sep. 2022.

[27] X. Huang, W. Wu, S. Hu, M. Li, C. Zhou, and X. Shen, "Digital twin based user-centric resource management for multicast short video streaming," 2023, *arXiv:2308.08995*.

[28] H. Xu, J. Wu, Q. Pan, X. Liu, and C. Verikoukis, "Digital twin and Meta RL empowered fast-adaptation of joint user scheduling and task offloading for mobile industrial IoT," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 10, pp. 3254–3266, Oct. 2023.

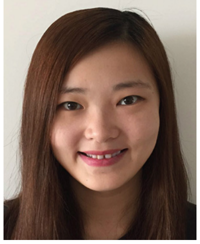
[29] D. Anick, D. Mitra, and M. M. Sondhi, "Stochastic theory of a data-handling system with multiple sources," *Bell Syst. Tech. J.*, vol. 61, no. 8, pp. 1871–1894, Oct. 1982.

[30] R. D. Yates and S. K. Kaul, "The age of information: Real-time status updating by multiple sources," *IEEE Trans. Inf. Theory*, vol. 65, no. 3, pp. 1807–1827, Mar. 2019.

[31] Y. Wang, W. Yin, and J. Zeng, "Global convergence of ADMM in nonconvex nonsmooth optimization," *J. Sci. Comput.*, vol. 78, pp. 29–63, Jan. 2019.

[32] G. França, D. P. Robinson, and R. Vidal, "A nonsmooth dynamical systems perspective on accelerated extensions of ADMM," *IEEE Trans. Autom. Control*, vol. 68, no. 5, pp. 2966–2978, May 2023.

[33] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. 2nd Int. Conf. Learn. Represent. (ICLR)*, 2014, pp. 1–14. [Online]. Available: http://arxiv.org/abs/1312.6114



Mushu Li (Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2021.

She was a Postdoctoral Fellow with Toronto Metropolitan University, Toronto, ON, Canada, from 2022 to 2024 and the University of Waterloo from 2021 to 2022. She is currently an Assistant Professor with the Department of Computer Science, Lehigh University, Bethlehem, PA, USA. She was the recipient of the Natural Science and Engineering Research Council of Canada Postdoctoral Fellowship in 2022, the NSERC Canada Graduate Scholarship in 2018, and the Ontario Graduate Scholarship in 2015 and 2016. Her research interests include mobile edge computing, the system optimization in wireless networks, and machine learning-assisted network management.

Jie Gao (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in electrical engineering from the University of Alberta, Edmonton, AB, Canada, in 2009 and 2014, respectively.

He was a Postdoctoral Fellow with Toronto Metropolitan (formerly Ryerson) University, Toronto, ON, Canada, from 2017 to 2019 and a Research Associate with the University of Waterloo, Waterloo, ON, Canada, from 2019 to 2020. He was an Assistant Professor with the Department of Electrical and Computer Engineering, Marquette University, Milwaukee, WI, USA, from 2020 to 2022 and is currently an Assistant Professor with the School of Information Technology, Carleton University, Ottawa, ON. His research interests include machine learning for communications and networking, cloud and multiaccess edge computing, Internet of Things and industrial IoT solutions, and B5G/6G networks in general.

Conghao Zhou (Member, IEEE) received the B.Eng. degree from Northeastern University, Shenyang, China, in 2017, the M.Sc. degree from the University of Illinois at Chicago, Chicago, IL, USA, in 2018, and the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2022.

He is currently a Postdoctoral Fellow with the University of Waterloo. His research interests include space-air-ground integrated networks, multitier computing, and machine learning in wireless networks.

Lian Zhao (Fellow, IEEE) received the Ph.D. degree in department of electrical and computer engineering from the University of Waterloo, Canada, in 2002.

She is currently a Professor with the Department of Electrical, Computer, and Biomedical Engineering, Toronto Metropolitan University, Toronto, ON, Canada.

Prof. Zhao received the Best Land Transportation Paper Award from IEEE VTS in 2016 and the Canada Foundation for Innovation New Opportunity Research Award in 2005. She served as an Industry Panel Co-Chair for IEEE Globecom 2024, the General Chair for the 2023 ComSoc Frontier Networking Symposium, and the Co-Chair for the Wireless Communication Symposiums for IEEE Globecom, ICC, and PIMRC. She has been serving as an Editor for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE INTERNET OF THINGS JOURNAL, and IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY from 2013 to 2021 and the Associate Editor-in-Chief for *China Communications*. She has been an elected member of the Board of Governors for VTS 2023 to 2025. She is a Professional Engineer in the Province of Ontario. She has been an IEEE Communication Society (ComSoc) and IEEE Vehicular Technology Society Distinguished Lecturer.

Xuemin (Sherman) 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, 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, 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 2016, IEEE Infocom 2014, IEEE VTC 2010 Fall, IEEE Globecom 2007, and the Chair for the IEEE Communications Society Technical Committee on Wireless Communications. He served as the Editor-in-Chief for IEEE INTERNET OF THINGS JOURNAL, IEEE NETWORK, and IET COMMUNICATIONS. He is the Past President of the IEEE ComSoc. He was the Vice President for Technical and Educational Activities, the Vice President for Publications, the Member-at-Large on the Board of Governors, the Chair of the Distinguished Lecturer Selection Committee, and a Member of IEEE Fellow Selection Committee of the ComSoc. He is a registered Professional Engineer of Ontario, Canada, an Engineering Institute of Canada Fellow, a Canadian Academy of Engineering Fellow, a Royal Society of Canada Fellow, a Chinese Academy of Engineering Foreign Member, and a Distinguished Lecturer of the IEEE Vehicular Technology Society and Communications Society.