

Received August 17, 2021, accepted September 4, 2021, date of publication September 16, 2021, date of current version October 5, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3113298

# Recurrent Neural Network-Based Optimal Sensing Duty Cycle Control Method for Wireless Sensor Networks

SEUNG-HEE CHOI<sup>ID</sup> AND SANG-JO YOO<sup>ID</sup>, (Member, IEEE)

Department of Electrical and Computer Engineering, Inha University, Incheon 402-751, South Korea

Corresponding author: Sang-Jo Yoo (sjyoo@inha.ac.kr)

This work was supported by Inha University Research Grant.

**ABSTRACT** With the development of Internet of Things (IoT) technologies, environmental monitoring systems using wireless sensor networks (WSNs) have received considerable attention. Reliable object detection and tracking is an important research issue in various WSN applications, such as environment and disaster monitoring, disaster propagation tracking, and intruder monitoring and tracking. Generally, because batteries are used as energy sources for sensors in WSNs, a highly energy-efficient operation is needed to prolong the life of the sensors and networks. To save energy, sensors usually manage multi-mode sensing operations, in which they periodically switch between active and inactive periods. A tradeoff exists between object detection accuracy and energy efficiency when we select a sensing schedule. Depending on the object speed, direction, and sensor deployment topology, different sensing schedules should be dynamically applied to individual sensors. In this paper, we propose a novel recurrent neural network (RNN)-based dynamic duty cycle control method for sensor nodes. For RNN training, a target optimal duty cycle for a given network condition is derived from the proposed digital twin-space analytic solution. Simulation results show that the proposed model provides accurate object detection performance and achieves high energy efficiency.

**INDEX TERMS** Duty cycle control, machine learning, object tracking, recurrent neural network, wireless sensor networks.

## I. INTRODUCTION

As the Internet of Things (IoT) and wireless sensor networks (WSNs) are becoming a reality, their interconnections for smart devices are increasing [1], [2]. IoT technologies will be able to connect physical things through sensors, actuators, and networks, and then control them. Hence, by enabling easy access and interaction with a wide variety of devices, such as home appliances, surveillance cameras, humidity sensors, actuators, forest fire detection sensors, vehicle trackers, and mobile phones, many objects surrounding us will be connected to networks in one form or another. In WSNs, sensor nodes collect data from the target environment, and deliver the data to the sink node (e.g., a server) through an ad-hoc sensor network and existing Internet infrastructure. Object monitoring and tracking is one of the main functionalities that

can be applied to many application areas. An object tracking sensor network monitors both indoor and outdoor environments and tracks various objects, such as forest fires, polluted air, bio-chemical materials, automobiles, and animals [3]–[6]. WSN sensor nodes can monitor combustion gases and preemptive fire conditions to define alert zones. In addition to preventive measures, early detection of fires is the only way to minimize damage and casualties. Pollution monitoring and display to citizens are essential to compare the impact of measures taken by municipalities and public institutions, and to raise public awareness. WSNs are also widely used for wildlife monitoring and tracking of different species. For intrusion detection, autonomous sensors equipped with video cameras enable the development of new security, surveillance, and military applications.

In WSNs, sensor nodes depend on limited battery capacity, and it is generally unrealistic to supplement or replace sensors in real-world applications [7], [8]. Thus, it is very important

The associate editor coordinating the review of this manuscript and approving it for publication was Fakhru Alam<sup>ID</sup>.

to increase the energy efficiency of the sensors needed, in order to prolong the life of the sensors and networks. Different methods for sensor operation in terms of sensor-activation scheduling have been studied. The most basic method is naïve activation, which always turns on the sensors [9]. In this case, the energy consumption of the sensor increases, and the lifetime of the network and sensor is shortened. Random activation is a method of operating sensors stochastically, which has the disadvantage that object tracking may not be smooth. Selective activation is a method of predicting the location of the target and activating only a few sensors around the target in the tracking mode. It has the advantage of saving more energy compared to other methods. The duty-cycle operation activates sensors at a constant time interval. The ratio of the active period to the sensing interval is called the duty cycle [10], [11]. Many studies are being conducted on how to activate sensors around objects with duty-cycled activation to efficiently use energy. There is a problem in that maximizing the inactive period to save energy causes severe data transmission delays and prevents the network from performing proper object detection functions [12], [13]. In object tracking of WSNs, it is very important to dynamically determine the optimal sensing duty cycle. The accuracy of object tracking and energy conservation is a trade-off relationship. A higher duty cycle results in a better object detection performance but lower energy efficiency.

The optimal sensing schedule (i.e., duty cycle) of each sensor node depends on the object's moving speed and direction, movement pattern, neighbor sensor node deployment topology, and object detection requirements. An individual sensor node is unable to properly change its sensing duty cycle in advance based only on its measurement because it does not know whether the object is approaching or moving away from itself until the object enters the sensing coverage of the node. Furthermore, even a server cannot easily calculate the optimal duty cycle for each node in real time with measurement information from sensors.

In this paper, we propose a novel recurrent neural network (RNN)-based optimal duty cycle control method for WSNs, in which the optimal sensing schedule is dynamically determined for each sensor node. We use an RNN learning structure because it can capture the sequential and temporal dynamic behavior well [14]. Based on the neighbor node's object detection results, in the proposed method, the server determines the node that must change its duty cycle and defines the optimal duty cycle that can meet the detection requirements with minimum energy consumption. We define multiple discrete duty cycle operation modes. In our long short-term memory (LSTM)-based RNN architecture, the input is defined as the network-wise sensor deployment topology and each sensor node's object observation history. The output of the RNN model is the optimal duty cycle of each sensor node. The proposed model can achieve efficient energy consumption with high object-tracking accuracy.

The main contributions of this paper can be summarized as follows:

- We derive an optimal duty-cycle computing method to obtain the target values (i.e., labels) for RNN-based supervised learning. For the ideal condition, in which object movements are known in advance, the optimal duty cycle of each node that satisfies the object tracking requirements is calculated.
- We implement a digital twin model to generate a training dataset for the RNN model, and to obtain the corresponding label set (i.e., for the given condition, the corresponding optimal duty cycle for each node).
- Using the data generated from the digital twin space, we perform learning for the proposed RNN model, and confirm the performance in terms of object detection and energy efficiency.

The remainder of this paper is organized as follows. In Section II, we discuss related work. The proposed RNN-based duty cycle control method for object tracking is presented in Section III. In Section IV, the simulation results are presented, and we conclude this paper in Section V.

## II. RELATED WORK

Several approaches for object tracking in WSNs have been proposed in recent years. In [15], Luo *et al.* proposed a cooperative target localization and tracking algorithm to reliably track objects in indoor wireless sensor networks, in which they considered and collaborated with the different characteristics of multiple networks. In [16], Lui *et al.* proposed a diffusion-distance-based predictive tracking algorithm for continuous objects such as fire and gas in industrial WSNs (IWSNs). They predicted the spread range of continuous objects by establishing the relationship between the diffusion radius and time, based on the assumption that the motion of continuous objects follows the appropriate diffusion model. Mohajerzadeh *et al.* proposed an algorithm for tracking mobile targets using directional sensor networks (DSNs) [17]. Unlike omnidirectional sensors, DSNs are networks consisting of directional sensors, such as image sensors, video sensors, and infrared sensors. As such, object tracking using WSNs is being studied for various environments, including indoor or outdoor, and various types of sensor situations.

Studies on how to efficiently control the sensing schedule of sensors have been conducted to reduce the energy consumption of sensors. Zhang *et al.* [18] proposed a new sleep scheduling method based on the decentralized partially observable Markov decision process and a sleep scheduling algorithm in online planning. They set up compensation for sleep scheduling problems by considering four factors: coverage, connectivity, tracking, and energy cost. In addition, the method makes decisions optimized for the entire cluster rather than for individual sensors. Jiang *et al.* [19] presented a probability-based prediction and sleep scheduling protocol to improve energy efficiency in

TABLE 1. Symbols used in this paper.

Symbol	Description	Symbol	Description
$K$	Number of discrete duty-cycle modes	$(x^i, y^i)$	Sensor $i$ 's location
$i$	Sensor node index	$(x^o, y^o)$	Object's location
$T_{wake}$	Fixed sensor active period	$r_t^i$	Sensor $i$ 's object detection result at $t$
$T_{sleep}^k$	Sensor's inactive period in mode $k$	$\delta$	Sensing interval margin used for LSTM model training
$T_{mode k}$	Sensing cycle time (sensing interval) of mode $k$	$t_{SI}^{i,T}$	Maximum sensing interval of node $i$ for training
$d$	Uncertain area radius	$k_t^{i,T}$	Selected optimal duty cycle mode of node $i$ for training at $t$
$t_{in}^i$	Time an object enters the sensing area of sensor $i$	$\hat{k}_t^i$	Output of RNN (derived duty cycle mode) of node $i$ at $t$
$t_{out}^i$	Time an object leaves the sensing area of sensor $i$	$D$	Input dataset for LSTM RNN learning
$t_{pass}^i$	Actual object passing time of the sensing area of sensor $i$	$n_d$	Number of datasets for LSTM RNN learning
$t_{min}$	Predefined minimum passing time for object tracking	$n_s$	Number of sensors in a training field
$N_{min}$	Required number of detections when $t_{pass}^i < t_{min}$	$l$	LSTM sequence length
$N_{req}$	Required number of detections when $t_{pass}^i \geq t_{min}$	$V_{max}$	Predefined maximum speed of an object
$t_{req}$	Maximum detection time interval for object tracking	$V_{min}$	Predefined minimum speed of an object
$N^i$	Minimum required number of detection times for node $i$	$P_s$	Sensing power
$t_{SI}^i$	Maximum sensing interval of node $i$	$E_k$	Sensing energy for mode $k$
$k_*^i$	Selected optimal duty-cycle mode of node $i$	$e_k$	Energy efficiency of mode $k$
$t_{obs}$	Observation interval	$N_t^{i,RNN}$	Number of detection times using RNN output for node $i$ at $t$

single-object tracking. They not only predict the next position of the object based on kinematics and probability but also derive the probability of moving along all directions; based on this, the nodes to be activated are selected, and the active time is controlled. Medagliani *et al.* [20] proposed an analytical framework that can maximize the lifetime of a network using a duty cycle method that considers the probability of missing object detection, detection delay, transmission delay, and average energy consumption. All nodes in the network use the same duty cycle. Kim *et al.* [21] proposed a mobility-aware adaptive duty-cycling mechanism for tracking objects during tunnel excavation. It tunes the duty-cycle ratio by adjusting the sleep time depending on the changes in the received signal strength indication (RSSI) value.

Recently, as research on machine learning has become more active, various studies have been conducted to combine machine learning to improve WSN performance [22], [23]. Na and Yoo [24] dynamically derived unmanned aerial vehicle (UAV) optimal locations using a particle swarm optimization (PSO)-based bio-inspired algorithm for collecting sensor data from WSNs. Yun and Yoo [25] proposed a Q-learning-based data-aggregation-aware energy-efficient routing algorithm for WSNs. They showed that it effectively reduces the amount of duplicated data and extends the lifetime of the network. Wei *et al.* [26] proposed an RNN-based delay-guaranteed monitoring framework in underwater WSNs considering delay, energy, and data quality to solve the problem of longer delay times for packet retransmission due to the high data loss rate during underwater acoustic transmission. Mohanti *et al.* [27] proposed a deep learning-based distributed data mining (DDM) model for energy efficiency and optimal load balancing, which reduces the energy consumption, signaling overhead, and average

delay, and maximizes the overall throughput compared to other methods.

Most research on WSNs is limited to certain situations and applications; thus, it is often difficult to apply flexibly to other situations. Furthermore, there is a lack of machine learning-based research on adaptively controlling the duty cycle according to the movement of objects and sensor node deployment topology and efficiently using the energy of sensors in WSNs simultaneously.

Unlike the existing object tracking methods in WSNs that require accurate mathematical target object location estimation, movement speed, direction estimation, or information on the sensing operation statistics of sensors, the method proposed in this paper requires only the sensors' location information and their object detection history for actual object tracking in real environments. The optimal duty cycle of each sensor is obtained using the proposed RNN learning architecture with digital twin logical space solutions that reflect various object movements and sensor network topology situations.

### III. PROPOSED RNN-BASED DYNAMIC DUTY CYCLE CONTROL METHOD FOR OBJECT TRACKING

#### A. SYSTEM MODEL

We propose an RNN-based system model that dynamically controls the activation schedule of each sensor to accurately detect objects and to use energy efficiently in situation, where one or more objects move freely in a WSN area. The symbols presented in Table 1 are used in the remainder of this paper.

The structure of the proposed system model is shown in Fig. 1. The proposed system consists of a digital twin model and a learning model. A digital twin is a virtual world that is implemented on a computer. It enables the simulation

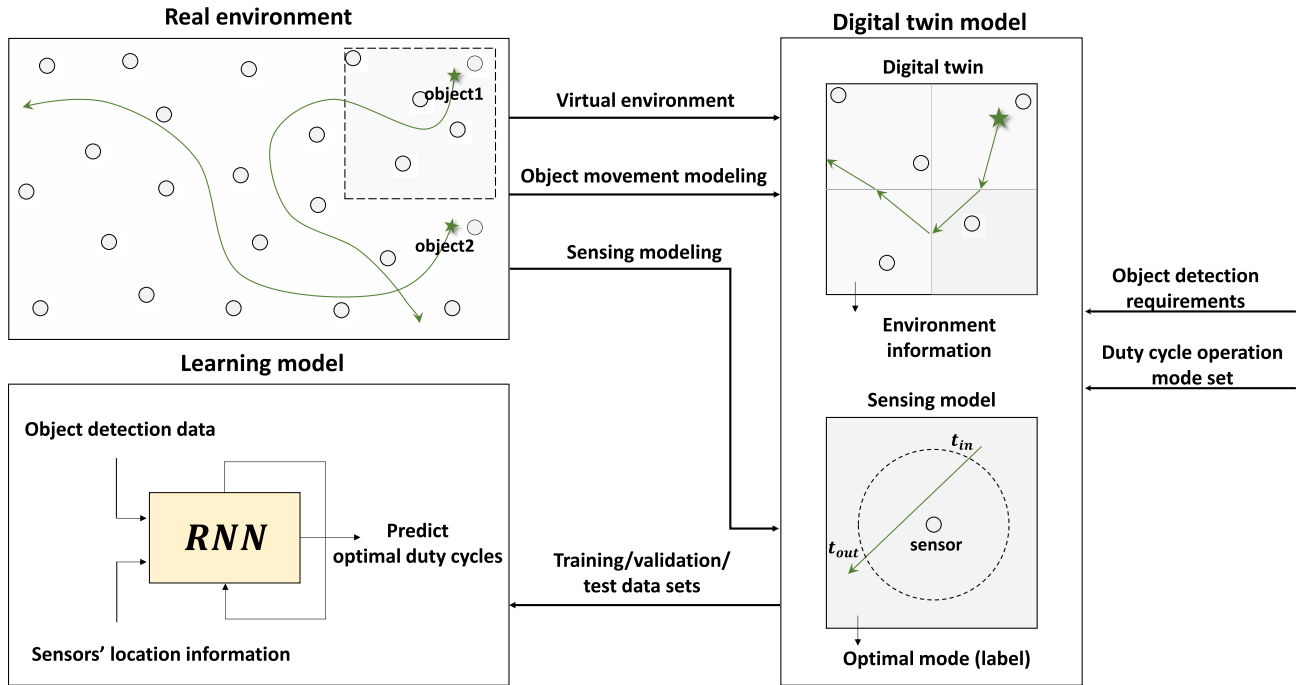


FIGURE 1. The proposed system model structure.

of environments, machines, equipment, and objects in the real world by implementing them in virtual space. A digital twin can efficiently predict the effects of product and process development without the need to create a costly and time-consuming physical model. In this study, a digital twin model is used to derive the set of sensing situation data and target labels (the optimal duty cycle modes for the corresponding sensing situations) in various environments necessary for learning the RNN-based model. As shown in Fig. 1, the sensor field environment, object movement, and sensing capability are modeled and implemented in the digital twin space; it also considers object detection requirements, and a predefined number of sensing duty cycle modes. Because a large number of sensor nodes can be deployed in a large physical space in the entire sensor field, in order to consider only the correlated regions in object tracking for learning, instead of modeling the entire WSN field, we use a predefined limited-size area for learning, which is defined as the training sensor field in this study. To model object movements, existing movement models can be used, but for the limited-size training sensor field, relatively simple mobility models can be applied.

The optimal sensing duty cycle for each sensor node for various sensor topologies and object movement conditions in the digital twin space is calculated using the optimal duty cycle derivation method proposed in this paper. The optimal sensing duty cycle mode that satisfies the given sensing requirements is determined by considering the time for entering and leaving the sensing coverage of a specific sensor in consideration of the moving direction and speed of a moving object. At this time, the positions of the sensors and

their object detection results in accordance with the derived optimal duty cycle modes in the training sensor field are collected as a dataset for learning.

As shown in Fig. 1, the RNN-based learning model of the proposed system is trained using the sequential dataset and labels generated from the digital twin model. In a real WSN environment, the RNN model trained in this way can dynamically predict the most energy-efficient duty cycle mode that can meet the object detection requirements.

### B. PROPOSED OPTIMAL DUTY CYCLE DERIVATION METHOD FOR SUPERVISED LEARNING

In this section, we derive the optimal duty cycle mode for each sensor under the given condition, in which each sensor should satisfy object detection requirements in accordance with the movement of the object. The duty cycle is the ratio of the active sensing time to the active sleep cycle time (i.e., sensing interval) [28]. We assume that the sensors use a discrete number of duty cycles. In this paper, we define  $K$  sensor duty cycle operation modes from 0 to  $(K - 1)$ . The time duration for which a sensor is active is defined as  $T_{wake}$ , and it is assumed that it is the same for all modes;  $T_{sleep}^k$  is the sleep time duration of mode  $k$  when a sensor is not active. The sensing interval (i.e., the cycle time) for mode  $k$  is defined as  $T_{mode k}$ , which is the sum of the times  $T_{wake}$  and  $T_{sleep}^k$ . Therefore, when the sensor is in mode  $k$ , the duty cycle and  $T_{mode k}$  are defined as follows:

$$Duty\_cycle_k = \frac{T_{wake}}{T_{sleep}^k + T_{wake}} \times 100 \quad (1)$$

$$T_{mode k} = T_{sleep}^k + T_{wake} \quad (2)$$

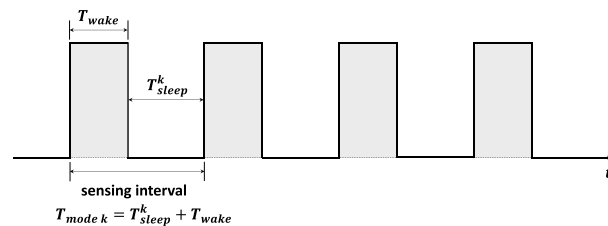
When the mode number  $k$  is closer to zero, it has a higher duty cycle. In contrast, if the mode number is closer to  $(K - 1)$ , it has a lower duty cycle. When the mode is zero, the sensor is always active. From mode 1 to mode  $(K - 1)$ , the sensor activation time  $T_{wake}$  is fixed to an appropriate value. In this study, we assume that objects can always be detected correctly if objects enter the sensing area when the sensor is activated.

Fig. 2(a) illustrates the definition of the duty cycle. We adjust the duty cycle by adjusting the length of  $T_{sleep}^k$  according to mode  $k$ . Thus, a longer  $T_{sleep}^k$  length results in a lower duty cycle, which results in a mode closer to  $(K - 1)$ . It should be noted that a higher duty cycle (i.e., lower mode number) requires more energy for sensor operation, but it can achieve higher object detection and tracking performance. We define three different types of areas, as shown in Fig. 2(b). The coverage where a sensor can detect objects is called the “sensing area”. Sensors should satisfy object detection requirements when objects enter their sensing area. We call the area where a sensor cannot detect objects because the objects are located outside the sensing area of the sensor, but are within a predefined distance  $d$  from the sensor as an “uncertain area”. In this area, even though the objects are outside the sensing coverage of the sensor, the sensor should be prepared for object detection because the objects may enter the sensing area in a short time. The outside of the uncertain area is called the “outer area”. In this case, the objects are not nearby, so that the sensor can maintain the lowest duty cycle. The relationship between  $T_{mode k}$  and duty cycle according to the location of the objects is shown in Fig. 2(c).

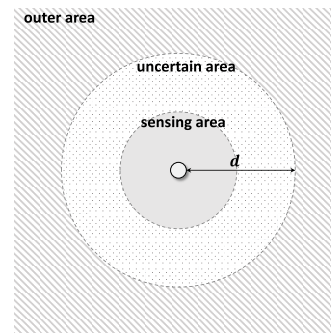
If an object is in the outer area, the lowest duty cycle  $T_{mode (K-1)}$  is selected as the optimal sensor activation cycle. If an object is in an uncertain area, in which the object is outside of the sensing area but within a predefined constant distance  $d$ , the sensor needs to change to the next higher duty cycle,  $T_{mode (K-2)}$  to properly detect the object when it enters the sensor’s coverage. Eventually, when an object enters the sensing area, the optimal duty cycle should be dynamically determined to satisfy the predefined object detection and tracking requirements. The optimal duty cycle mode of each sensor node depends on the locations of the object and sensors, the speed and movement direction of the object, the sensor coverage, and the detection and tracking requirements. When an object passes through the sensing area of a sensor node  $i$ , the exact entry and exit times to and from the sensing area are defined as  $t_{in}^i$  and  $t_{out}^i$ , respectively. Therefore, the object passing time of sensor  $i$ ’s sensing area is denoted as  $t_{pass}^i = t_{out}^i - t_{in}^i$ .

In this paper, we define the object detection and tracking requirements as follows.

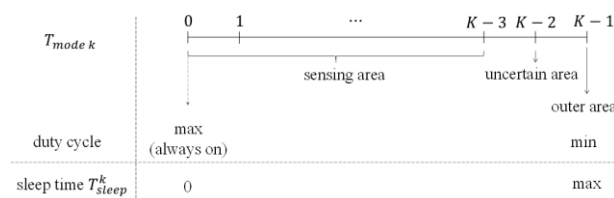
- i) *Minimal detection count requirement 1*: If the object passing time  $t_{pass}^i$  is less than  $t_{min}$ , then the sensor should detect the object at least  $N_{min}$  times during the object passing time.
- ii) *Minimal detection count requirement 2*: If the object passing time  $t_{pass}^i$  is greater than or equal to  $t_{min}$ , then



(a)



(b)



(c)

**FIGURE 2. (a) Duty cycle definition. (b) Area definitions. (c) Relationship between area definitions,  $T_{mode k}$ , and duty cycle.**

the sensor should detect the object at least  $N_{req}$  times during the object passing time.

- iii) *Minimal detection interval requirement*: As long as the object stays within the sensing area, the sensor should detect the object at least at every  $t_{req}$  time.

The terms  $t_{min}$ ,  $N_{min}$ ,  $N_{req}$ , and  $t_{req}$  are predefined in accordance with the object detection and tracking objectives for different WSN application scenarios.

The maximum sensing interval when an object enters the sensing area can be determined as  $T_{mode (K-3)}$  to satisfy requirement (iii) as

$$T_{mode (K-3)} = t_{req} \tag{3}$$

If the moving speed of an object is too fast or passes close to the boundary of the sensing area, then the object may stay for a shorter time in the sensing area than  $t_{min}$ . In this case, the sensor should detect the object at least  $N_{min}$  times to satisfy requirement (i). Otherwise, the sensor duty cycle should be able to detect the object at least  $N_{req}$  times to satisfy requirement (ii). The minimum required number of detection times  $N^i$  for node  $i$  can be represented as in (4).

$$N^i = \begin{cases} N_{min}, & \text{if } t_{pass}^i < t_{min} \\ N_{req}, & \text{if } t_{pass}^i \geq t_{min} \end{cases} \tag{4}$$

For any node  $i$ , in which the object is in its sensing area, the maximum sensing interval ( $t_{SI}^i$ ), to meet all requirements is derived as

$$t_{SI}^i = \frac{t_{out}^i - t_{in}^i}{N^i} = \frac{t_{pass}^i}{N^i} \quad (5)$$

Therefore, the optimal duty cycle mode  $k_*^i$  for node  $i$  among the predefined discrete duty cycle modes can be derived as in (6).

$$k_*^i = \begin{cases} 0 & \text{if, } t_{SI}^i < T_{mode 1} \\ m & \text{if, } T_{mode m} \leq t_{SI}^i < T_{mode(m+1)} \end{cases} \quad (6)$$

The derived optimal duty cycle mode  $k_*^i$  guarantees not only satisfying the object detection and tracking requirements but also selecting the most energy-efficient sensing mode operation. When the mode is determined according to this formula, the sensor satisfies the detection requirements as follows:

$$\frac{t_{out}^i - t_{in}^i}{T_{mode k_*^i}} \geq N^i \quad (7)$$

Fig. 3 shows the proposed optimal sensing schedule derivation scenarios for different object speeds and directions. Fig. 3(a) shows the case where an object passes the boundary of the sensing area very fast, such that the derived optimal duty cycle mode is mode 0 (always active). The object passing time in Fig. 3(b) is shorter than that in Fig. 3(c), so that the duty cycle in Fig. 3(b) is higher than that in Fig. 3(c). The object moving speed in Fig. 3(d) is very slow, so that

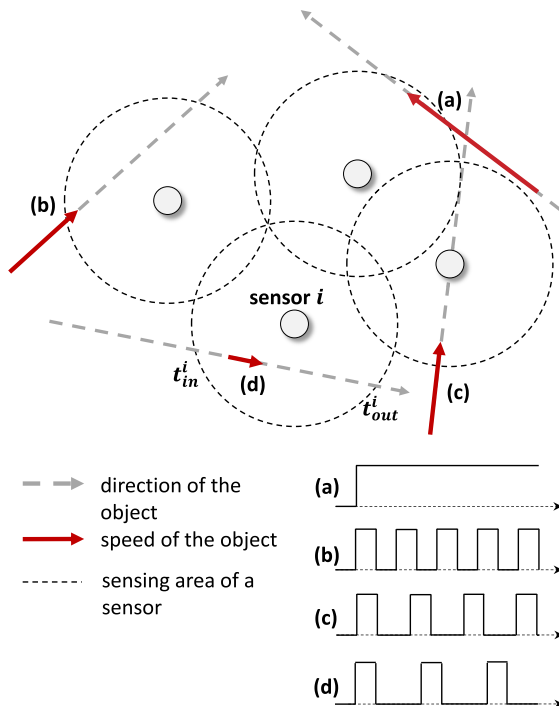


FIGURE 3. The proposed optimal sensing schedule derivation scenarios.

the selected duty cycle is the lowest. In all cases, the derived duty cycle modes guarantee the object detection and tracking requirements.

Therefore, in the digital twin space, the optimal duty cycle of each sensor is determined in various sensor arrangement topologies and object movement situations, and these data are used to train the proposed RNN model in the next section. In the digital twin environment, each sensor operates in the derived optimal duty cycle mode to save energy, and at each unit time step ( $t_{obs}$ , observation interval), the sensing results (detection or non-detection) of all sensors are stored together with the positions of the sensors. In addition, the derived optimal modes of the sensors are stored as labels for RNN learning.

### Algorithm 1: Optimal Duty Cycle Derivation and RNN Learning Method

#### A. Digital Twin Simulation

**Input:** sensor network virtual environment  $Env()$ , object movement model  $Move()$ , sensing areas, sensing requirements ( $N_{req}, N_{min}, t_{min}, t_{req}$ ),  $K$  duty cycle operation modes, training margin  $\delta$

**Output:** Set of object detection traces as a form of (sensor location, binary detection result, optimal duty cycle mode for training)

```

for different (topology, number_of_objects) do
    generate traces  $\sim Env(), Move()$ 
    for each unit_step  $t$  of a trace do
        for each sensor  $i$  do
            if object is in the sensing area then
                compute the maximum sensing interval  $t_{SI}^{i-T}$  in (9)
                derive the optimal duty cycle mode  $k_i^{i-T}$  in (10)
            else if object is in the uncertain area then
                 $k_i^{i-T} = (K - 2)$ 
            else object is in the outer area then
                 $k_i^{i-T} = (K - 1)$ 
            end if
            operate the sensor with mode  $k_i^{i-T}$ 
            obtain the binary object detection result  $r_t^i$ 
            store  $(x_t^i, y_t^i, r_t^i, k_i^{i-T})$  in replay memory  $\mathcal{M}$ 
        end for
    end for
end for

```

#### B. LSTM-based RNN Learning

**Input:** Replay memory  $\mathcal{M}$ , RNN sequence length  $l$

**Output:** LSTM-based RNN weight parameter  $\theta$

Initialize the RNN weight parameter  $\theta$

```

while not at end of training do
    sample random  $(x_t^i, y_t^i, r_t^i, k_i^{i-T})$  from replay memory  $\mathcal{M}$ 
    if exit  $(t - l + 1) \sim t$  data trace then
        make an RNN input data in (11) using all sensor's
        stored data in replay memory  $\mathcal{M}$  from  $(t - l + 1)$  to  $t$ 
        make an RNN output target data in (12) using all sensor's
        optimal duty cycle modes for training at time  $t$ 
    end if
    obtain the output  $\hat{k}_t^i (i = 1, \dots, n_s)$  of RNN
    update RNN weight parameter  $\theta$  to minimize cross
    entropy loss function  $L(\theta)$ 
end while

```

### C. RNN LEARNING MODEL

The proposed optimal duty-cycle decision method described in Section 3(B) is computed under the assumption that we know the exact time when an object enters and exits the sensing area. However, in a real environment, it is difficult to calculate the optimal duty cycle by predicting the exact moving trajectory and time because only the object detection results observed by each sensor can be known. Therefore, in this study, we use deep learning to predict the optimal duty cycle using sequential data containing object detection results. RNN is a class of artificial neural networks in which connections between nodes form a directed graph along a temporal sequence; this allows it to exhibit temporal dynamic behavior. Vanilla RNNs suffer from the problem of vanishing gradients when it has long data sequences, which results in poor learning [29], [30]. The LSTM was designed to alleviate this vanishing gradient problem. In this study, we predict the optimal duty cycle using an LSTM-based RNN architecture.

Because LSTM-based RNN uses supervised learning, it requires a true label for the input. In our WSN environment, the label is the optimal duty cycle of each sensor for the given condition. In the digital twin space, the object detection results of sensors, in which sensors operate with their optimal modes, implicitly reflect the object movements in the WSN. In RNN-based training, we use the location information of the sensors and object detection results for predefined time steps (RNN sequence length) as input data. The output label of the proposed RNN model corresponds to the optimal duty cycle of the sensors for the given input data.

One time step in an RNN corresponds to the observation interval ( $t_{obs}$ ) for reporting the sensing result at a sensor to the server, and for updating the overall detection status at the server. To generate the RNN training dataset, in the digital twin space, at every time  $t_{obs}$ , sensor node  $i$ 's position  $(x^i, y^i)$  and its object detection result  $r^i$  are stored, as shown in Fig. 4(a). The object detection result is represented as binary information (i.e., 0 is for "undetected," and 1 is for "detected"). In this study, we assume a simple binary sensor network [31]. Moreover, if we use more complex sensors that can obtain different resolutions of object detection (e.g., different detection values, such as the distance between the sensor and the object within the sensor coverage), then it results in a more accurate optimal duty cycle prediction in the RNN structure.

At each time step  $t$ , the input time step vector is defined as

$$d_t = \left[ \left( x_t^1, y_t^1, r_t^1 \right), \left( x_t^2, y_t^2, r_t^2 \right), \dots, \left( x_t^{n_s}, y_t^{n_s}, r_t^{n_s} \right) \right] \quad (8)$$

where  $x_t^i, y_t^i, r_t^i$  represent the  $x$  coordinate,  $y$  coordinate and object detection results of node  $i$  at time step  $t$ , respectively; and  $n_s$  is the number of sensors in the training field.

In this study, as the input of the RNN model, we only use the information that can be obtained with minimum constraints in real sensor network environments, sensor locations, and binary object detection results. This simple RNN input data type has the advantage of being generally

applicable to any scenario without complex calculations, additional functional modules, or any cooperative protocols with adjacent nodes in the real environment. However, it may be difficult to accurately predict the maximum sensing interval of (5), which is derived analytically with  $t_{in}^i$  and  $t_{out}^i$  estimation in a digital twin space. In this paper, although it may cause little additional energy consumption, the sensing interval margin  $\delta$  is introduced as in (9) to conservatively set the maximum sensing period of node  $i$  for training,  $t_{SI}^{i,T}$ . This allows the use of the actual sensing interval that is conservatively reduced by  $\delta$  ( $0 \leq \delta \leq 1$ ) from the ideal optimal maximum sensing interval that satisfies the object detection requirements. When an object enters the sensing area of node  $i$  at time step  $t$ , the target label  $k_t^{i,T}$  for RNN model training considering the sensing interval margin  $\delta$  is derived as in (10).

$$t_{SI}^{i,T} = t_{SI}^i (1 - \delta) \quad (9)$$

$$k_t^{i,T} = \begin{cases} 0 & \text{if, } t_{SI}^{i,T} < T_{mode\ 1} \\ m & \text{if, } T_{mode\ m} \leq t_{SI}^{i,T} < T_{mode\ (m+1)} \end{cases} \quad (10)$$

As we mentioned, at time step  $t$ , if the object is in the outer area, then  $k_t^{i,T} = (K - 1)$  is selected; if the object is in an uncertain area, then  $k_t^{i,T} = (K - 2)$  is determined as the target label.

As shown in Fig. 4(b), the input dataset  $D$  for the LSTM-based model with sequence length  $l$  and the corresponding target label  $T$  are defined as in (11) and (12), as shown at the bottom of the next page, respectively.

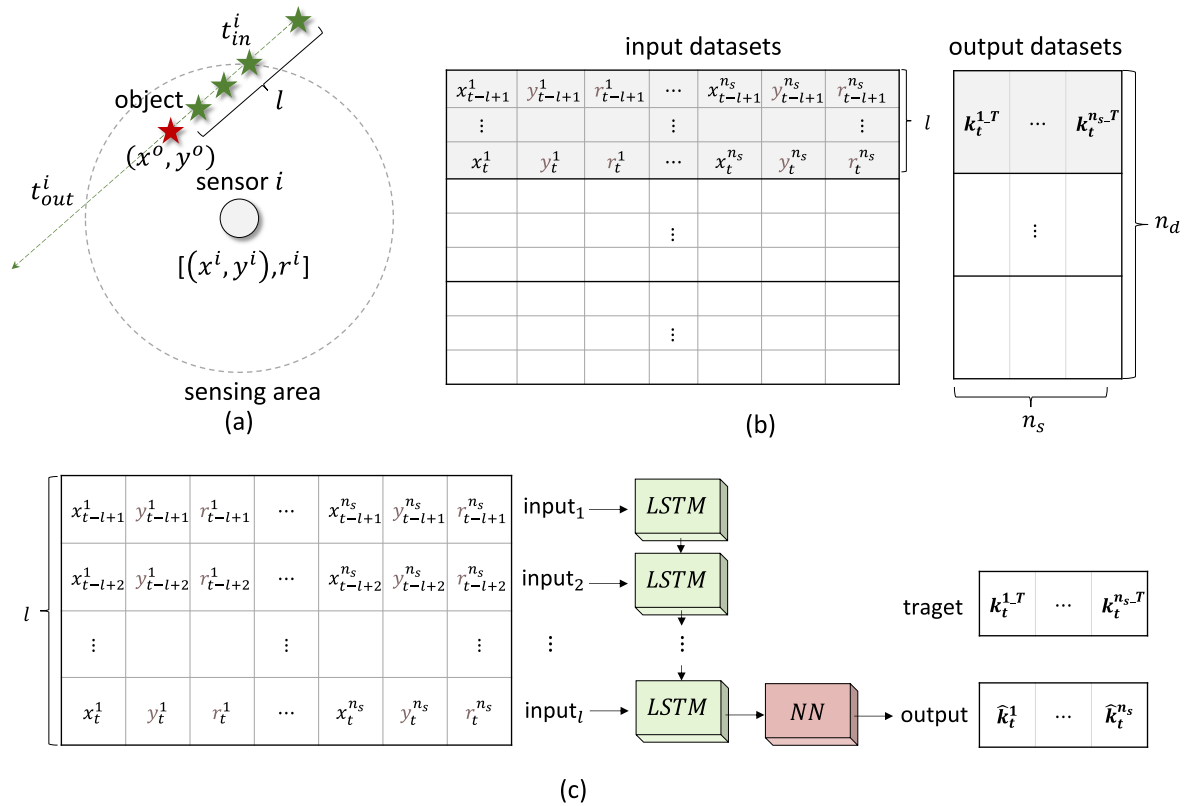
The target  $T$  vector indicates the optimal duty cycles at time step  $t$  for all sensor nodes based on the last  $l$  input data conditions from  $(t - l + 1)$  to  $t$ . For training,  $n_d$  number of datasets is used. Fig. 4(c) shows the LSTM-based RNN structure for learning the duty cycle derivation;  $\hat{k}_t^i$  is the output of the RNN for node  $i$  at time  $t$ .

The optimal duty cycle mode derivation and RNN learning algorithm proposed in this paper are presented in Algorithm 1. In the digital twin simulation, we obtained simulation traces to train the RNN model using different WSN sensor topologies and object movements. In LSTM-based RNN learning, we randomly sample data with sequence length  $l$  from the replay memory and update the RNN weight parameter  $\theta$  to minimize the cross-entropy loss function  $L(\theta)$ .

In real WSN environments, we apply the trained RNN model to dynamically determine the optimal duty cycle model for each sensor. The RNN input, sensor location information, and past object detection sequence are used. According to the RNN output, each sensor dynamically changes its duty cycle mode. If the server knows that the pre-arranged sensor topology or sensors do not move, then the initial sensor location information can be used as the RNN input information.

## IV. SIMULATION RESULTS

In this section, we evaluate and analyze the performance of the proposed RNN-based optimal duty cycle control method in terms of mode decision accuracy, energy



**FIGURE 4.** Dataset configuration. (a) Extracted information from digital twin. (b) LSTM dataset form. (c) Proposed LSTM-based RNN model.

efficiency, and object detection accuracy. We implemented the digital twin WSN sensing environment using MATLAB, and the LSTM-based RNN learning model using Python and Keras (an open-source neural network library).

The digital twin simulation parameters and values used in this study are listed in Table 2, in which the unit time and unit distance are used. We used a random-type grid topology for the WSN, in which sensor nodes were deployed in the form of a grid.

An example topology case is shown in Fig. 5, where 40 sensors are placed in a  $20 \times 20$  WSN training sensor field.

The sensing area radius and uncertain area radius are set to 5 and 10, respectively. The number of objects is 1 for single-object scenarios, and 2 for multi-object scenarios. To model the object movement, we implemented a modified random waypoint model in which mobile objects move randomly and freely without restrictions [32]. Unlike the original random waypoint model, in which objects begin by pausing for a fixed number of seconds, in the modified model, objects pause for a random time between 0 and 5 units. Then, the objects select a random destination within the simulation range and choose a random speed in a predefined range  $[V_{min}, V_{max}]$ .

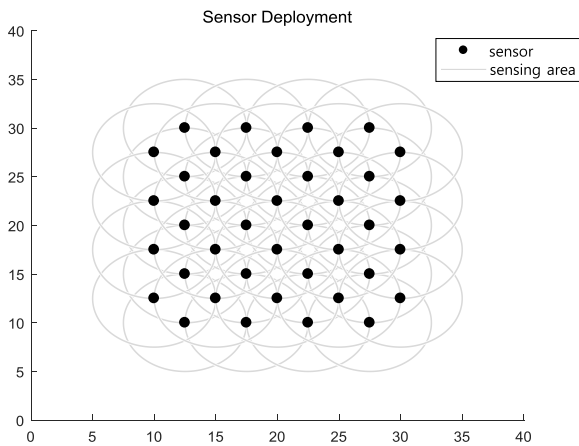
$$D = \begin{bmatrix} d_{t-l+1} \\ d_{t-l+2} \\ \vdots \\ d_t \end{bmatrix} = \begin{bmatrix} (x_{t-l+1}^1, y_{t-l+1}^1, r_{t-l+1}^1), \dots, (x_{t-l+1}^{n_s}, y_{t-l+1}^{n_s}, r_{t-l+1}^{n_s}) \\ (x_{t-l+2}^1, y_{t-l+2}^1, r_{t-l+2}^1), \dots, (x_{t-l+2}^{n_s}, y_{t-l+2}^{n_s}, r_{t-l+2}^{n_s}) \\ \vdots \\ (x_t^1, y_t^1, r_t^1), \dots, (x_t^{n_s}, y_t^{n_s}, r_t^{n_s}) \end{bmatrix} \quad (11)$$

$$T = \{k_t^{1-T}, k_t^{2-T}, \dots, k_t^{n_s-T}\} \quad (12)$$



**TABLE 2.** Simulation parameters for digital twin.

Parameter	Value	Parameter	Value
Training field size	$20 \times 20$	Sensing area radius	5
Object movement range	$30 \times 30$	Uncertain area radius $d$	10
Number of objects	1–2	Number of modes, $K$	6, 9
Number of sensors	40	Random pause time	[0,5]
Maximum object speed $V_{max}$	6	Minimum object speed $V_{min}$	4
Observation interval	0.25	$t_{min}$	0.3
$t_{req} = T_{mode(K-3)}$	1.0	$T_{wake}$	0.1
$\delta$	0.1	$N_{min}$	1
$N_{req}$	2		

**FIGURE 5.** Simulation environment: sensor deployment example.

Each object moves in the direction of the destination, and if it reaches the destination, it pauses for a random time and repeats the process. Objects move within a  $30 \times 30$  range that is slightly larger than the  $20 \times 20$  training area to capture the different mode selection scenarios for the boundary sensors. The observation interval to report the sensing results to the server was set to 0.25. The number of duty cycle modes ( $K$ ) were six and nine. In the lowest mode (mode 0), sensors are always active, so that they can detect objects all the time; however, they require the highest energy consumption. In the sensing area, the optimal mode between mode 0 and mode  $(K - 3)$  was selected. In the uncertain area and outer area, modes  $(K - 2)$  and  $(K - 1)$  were selected. The sensing interval margin  $\delta$  used for the LSTM model training was set to 0.1. This reduces the actual sensing interval by 10% from the ideal maximum sensing interval that satisfies the object detection requirements.

Table 3 lists the sensing intervals for different duty cycle modes when  $K = 6$  and 9. The hyperparameter values used in the LSTM implementation are listed in Table 4. Table 5 shows the distribution of datasets corresponding to the optimal duty cycle modes for 500000 datasets when the sequence length  $l$  was 10. Although 500000 datasets have been generated for learning in digital twin space, it is rare

**TABLE 3.** Sensing duty cycle modes ( $K = 6$  and 9).

$K = 6$		$K = 9$	
parameter	value	parameter	value
$T_{mode0}$	Always wake up	$T_{mode0}$	Always wake up
$T_{mode1}$	0.25	$T_{mode1}$	0.25
$T_{mode2}$	0.625	$T_{mode2}$	0.40
$T_{mode3}$	1.0	$T_{mode3}$	0.55
$T_{mode4}$	2.0	$T_{mode4}$	0.70
$T_{mode5}$	4.0	$T_{mode5}$	0.85
		$T_{mode6}$	1.0
		$T_{mode7}$	2.0
		$T_{mode8}$	4.0

**TABLE 4.** LSTM parameters.

Parameter	Value	Parameter	Value
LSTM layer	1	Batch size	128, 256
Dropout rate	0.2	Learning rate	0.01
Loss function	Cross-entropy	Sequence length	3,5,10,15
Training data	70%	Validation data	10%
Test data	20%	Total number of datasets	500000

for objects to move very fast and/or pass through the boundaries of sensing coverage. Moreover, most of the datasets are for scenarios where the objects are in the outer area of the sensors. There is a problem with unbalanced training data. Data imbalance refers to the situation where the classes in a dataset are not equally distributed, which can lead to potential risks in training a model. Therefore, it is necessary to adjust the balance between the datasets of different duty cycle modes. If the dataset is balanced based on the mode with the smallest number of datasets, then the RNN learning model may suffer from insufficient data for training. In this simulation study, to create balanced datasets, we used the under- and over-sampling method. However, the datasets for modes 0 and 1 were still smaller than others, even after balancing preprocessing. For the validation and test datasets, we did not perform any preprocessing for balancing. For performance evaluation, among the datasets created by the digital twin, the test datasets corresponding to 20% of the total datasets were used.

To evaluate the performance of the proposed LSTM-based RNN dynamic duty cycle mode decision method, two methods using fixed duty cycles were compared. The “compared\_model\_1” is a method in which the sensors use mode 1, in a fixed manner. Because mode 0 always maintains an active state, mode 1 is the mode with the highest duty cycle (i.e., the shortest sensing interval) among the modes that repeat the awake and sleep states. The “compared\_model\_2” is a method in which sensors always maintain duty cycle mode  $(K - 3)$ . Mode  $(K - 3)$  has the lowest duty cycle (i.e., the longest sensing interval) among the duty cycles used when the object is in the sensing area.

First, we evaluated the performance of “mode accuracy” for the cases of  $K = 6$  and  $K = 9$  with different

TABLE 5. Distribution of mode datasets (%).

	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6	Mode 7	Mode 8
Single object, K = 6	0.102	1.953	8.818	1.369	29.340	58.418			
Multi-objects, K = 6	0.214	4.038	16.885	2.536	42.459	33.869			
Single object, K = 9	0.101	0.357	0.851	1.872	4.428	3.273	1.356	29.336	58.427
Multi-objects, K = 9	0.214	0.747	1.757	3.809	8.664	6.048	2.441	42.477	33.843

RNN sequence lengths. In this paper, the mode accuracy was defined as a measure to evaluate how precisely the RNN outputs for the test datasets match the true optimal modes derived by (10). We define two mode-accuracy measures. First, “mode\_accuracy\_1” (MA1) is derived as in (13), and is the probability that the RNN output mode is the same with the true label.

$$MA1 = \left( \frac{1}{|TD| \times n_s} \sum_{t=1}^{|TD|} \sum_{i=1}^{n_s} \mathbb{I} \left( \text{output} \hat{k}_t^i, k_t^{i-T} \right) \right) \times 100 \tag{13}$$

where  $|TD|$  is the number of test datasets;  $\mathbb{I}(a, b)$  is the identity operator that returns 1 when  $a$  and  $b$  are identical; otherwise, it returns 0. Next, “mode\_accuracy\_2” (MA2) is defined as in (14), and shows how close the RNN output mode is derived to the true label.

$$MA2 = \left( \frac{1}{|TD| \times n_s} \sum_{t=1}^{|TD|} \sum_{i=1}^{n_s} \left( 1 - \frac{|\hat{k}_t^i - k_t^{i-T}|}{K} \right) \right) \times 100 \tag{14}$$

Fig. 6 shows the mode accuracy performance of the proposed method for two accuracy measures at different numbers of modes ( $K$ ), RNN sequence lengths ( $l$ ), and numbers of objects. As shown in Fig. 6(a), the proposed method achieves 81–92% mode\_MA1 accuracy. Specifically, for the single object and  $K = 6$  cases, the proposed method shows up to 92% accuracy. As shown in Fig. 6(b), MA2 for the proposed method is approximately 96–98%. This high performance is due to the fact that even though the mode classification of the RNN is not equal to the target, the misclassified mode has only one mode distance from the true value. For both mode accuracy measures, a smaller  $K$  ( $=6$ ) results in a better accuracy than that of a larger  $K$  ( $=9$ ). This is because the classifier accuracy using LSTM increases when the number of classes is small. The accuracy of the single-object tracking case was higher than that of the multi-object tracking case. We can also observe that as the RNN sequence length is increased, the accuracy performance tends to increase as well, except for some exceptional experimental results. This is because the longer the sequence length, the better the prediction of object movement.

The purpose of dynamic duty cycle control is to minimize the energy consumption while satisfying the object detection requirements. Therefore, the efficient use of energy is an important performance indicator. To evaluate the energy efficiency, we compared the energy required for each method

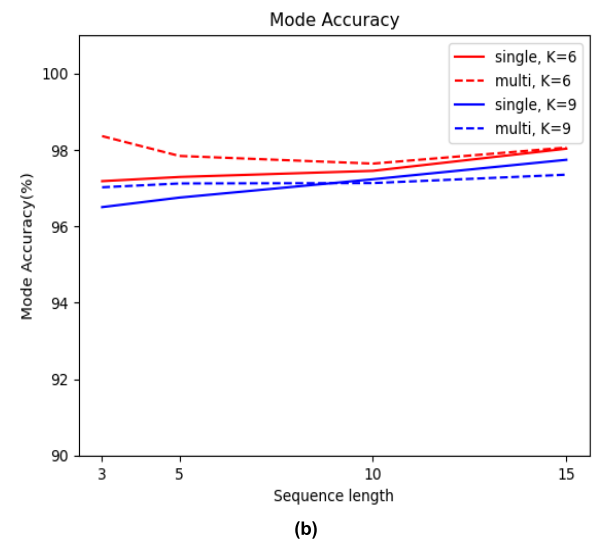
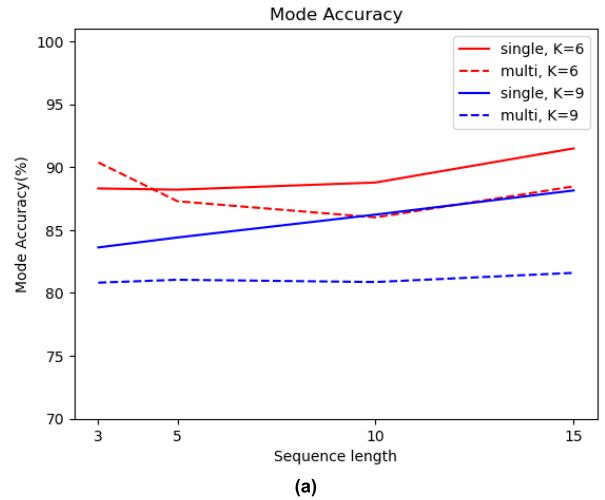


FIGURE 6. Mode accuracy. (a) MA1, and (b) MA2.

with the mode ( $K - 1$ ), which is the lowest duty cycle mode (i.e., the lowest sensing energy consumption mode). We considered only the energy required for sensing and did not include additional energy consumption for reporting the sensing results or exchanging control messages.

The energy consumed in a specific duty cycle mode  $k$  is  $E_k = \frac{T_{wake}}{T_{mode k}} P_s$ , where  $P_s$  is the sensing power. Therefore, the energy efficiency of mode  $k$  compared with that of mode ( $K - 1$ ) is defined as

$$e_k = \frac{E_{(K-1)}}{E_k} = \frac{T_{mode k}}{T_{mode (K-1)}} \tag{15}$$

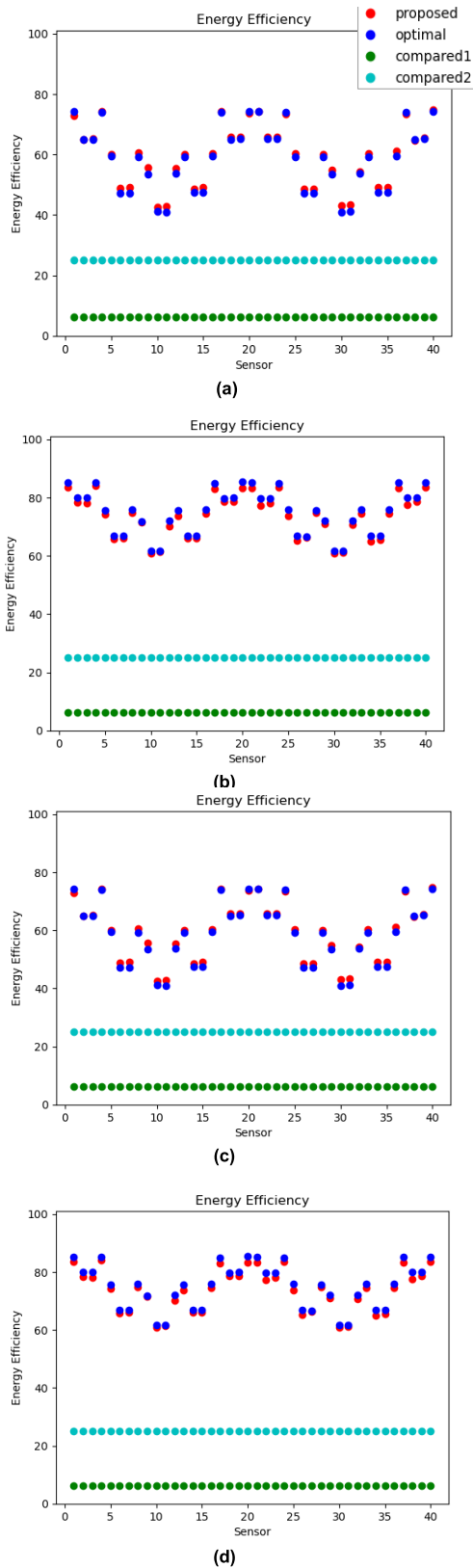


FIGURE 7. Energy efficiency. (a) Multi-object in  $K = 6$ . (b) Single object in  $K = 6$ . (c) Multi-object in  $K = 9$ . (d) Single object in  $K = 9$ .

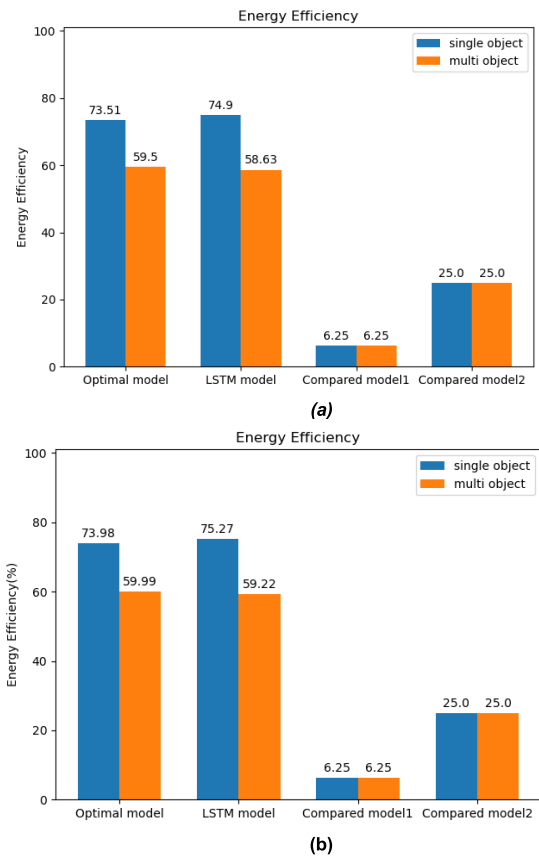


FIGURE 8. Energy efficiency. (a) Average in  $K = 6$ . (b) Average in  $K = 9$ .

In this paper, we define “energy\_efficiency” ( $EE$ ) for compared duty cycle methods, as in (16).

$$EE = \left( \frac{1}{|TD| \times n_s} \sum_{t=1}^{|TD|} \sum_{i=1}^{n_s} \left( \frac{T_{mode} y_t^i}{T_{mode} (K-1)} \right) \right) \times 100 \quad (16)$$

where  $T_{mode} y_t^i$  is the cycle time of the derived RNN output duty cycle mode,  $mode y_t^i$ , for node  $i$ .

Fig. 7 shows the energy efficiency of each sensor in the sensor field for different values of  $K$  (6 and 9), and different numbers of objects (1 and 2 for single- and multi-object cases). In the proposed method, the RNN sequence length was set to 10. In the graph, the red, blue, green, and turquoise points represent the results from the proposed model, optimal model, compared\_model\_1, and compared\_model\_2, respectively. The proposed LSTM-based RNN model shows an energy efficiency similar to that of the optimal model. For the multi-object cases, the energy efficiency of the proposed model is approximately 50% higher than that of compared\_model\_1 and is approximately 30–35% higher than that of compared\_model\_2. For single-object cases, the energy efficiency of the proposed method was

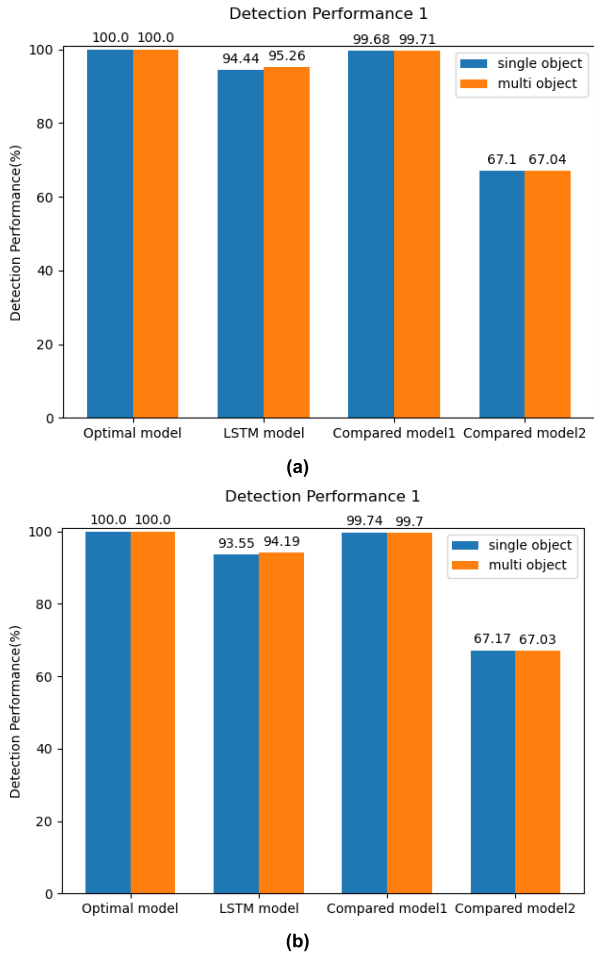


FIGURE 9. Detection performance DP1. (a)  $K = 6$ , and (b)  $K = 9$ .

approximately 70% higher than that of compared\_model\_1, and 50% higher than that of compared\_model\_2. In the proposed method, the energy efficiency of the multi-object case is lower than that of the single-object case, because as the number of objects increases, more objects enter the sensing area of the sensors and, accordingly, the sensors have to wake up more frequently.

Fig. 8 shows the average energy efficiencies of the compared models. The proposed LSTM-based RNN model shows an energy efficiency similar to that of the ideal optimal model that satisfies the detection requirements and operates in a mode that consumes the minimum sensing energy. The compared\_model\_1 operating with the highest duty cycle mode 1 in a fixed manner shows the lowest energy efficiency. We can also see that the energy efficiency of the proposed model is slightly higher for  $K = 9$  than for  $K = 6$ . This means that when the mode is more subdivided, the RNN model can determine the required duty cycle mode with high resolution, so that the energy can be saved slightly more than that in the lower resolution. However, the mode decision accuracy slightly decreases when a large  $K$  is used, as shown in Fig. 6.

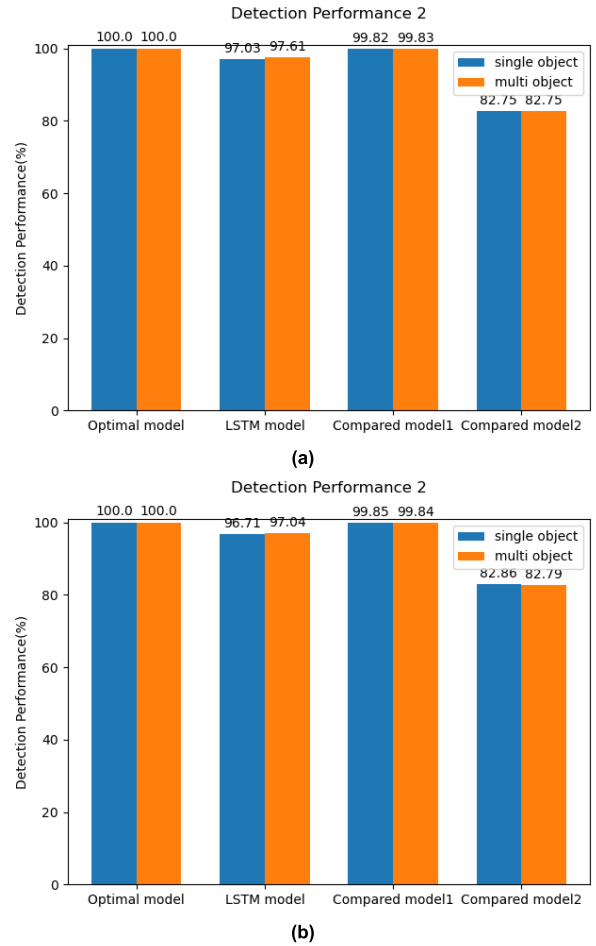


FIGURE 10. Detection performance DP2. (a)  $K = 6$ , and (b)  $K = 9$ .

An energy-efficient dynamic duty cycle mode operation must meet the object detection requirements. The optimal model always guarantees satisfying the object detection requirements, in which it is assumed that we perfectly know the object movement, so that we can compute (5) and (6). In the proposed LSTM-based RNN model, because we only use sequential binary sensing results from sensors, there is little decision error compared with the true optimal modes, as shown in Fig. 6. However, the mode decisions from the RNN model, which are different from the optimal modes, do not always result in an unsatisfactory object detection requirement. If the duty cycle mode from the RNN model is smaller than the optimal mode, then it has a shorter sensing interval, so that the energy efficiency decreases, but the object detection performance is better than the requirements.

In this simulation study, we defined two detection performance measures. First, “detection\_performance\_1” (DP1) is the percentage of satisfaction of the detection requirements, as expressed by (17).

$$DP1 = \left( \frac{1}{|TD| \times n_s} \sum_{t=1}^{|TD|} \sum_{i=1}^{n_s} \mathbb{I} \left( N_t^{i-RNN}, N_t^i \right) \right) \times 100 \tag{17}$$

where  $N_t^{i-RNN}$  is the number of object detection times using the derived duty cycle mode from the proposed RNN model for sensor  $i$  at the  $t$ -th test; and  $N_t^i$  is the exactly computed required object detection time using (4) for sensor  $i$  at the  $t$ -th test. Next, “detection\_performance\_2” (DP2) is defined as the average ratio of  $N_t^{i-RNN}$  to  $N_t^i$  for all test datasets, as expressed by (18).

$$DP2 = \left( \frac{1}{|TD| \times n_s} \sum_{t=1}^{|TD|} \sum_{i=1}^{n_s} \max \left( \frac{N_t^{i-RNN}}{N_t^i}, 1 \right) \right) \times 100 \quad (18)$$

We can see how close the number of detections by the selected mode is to the required number of detections with DP2.

Fig. 9 and Fig. 10 show DP1 and DP2, respectively, for  $K = 6$  and  $K = 9$ . In the proposed RNN model, the sequence length,  $l$ , is 10. For DP1 evaluation, the proposed method showed 93.55%–95.26% satisfaction with the required detection performance. In the proposed method, the DP1 performance for  $K = 6$  was slightly higher than that for  $K = 9$ . However, the difference was only approximately 1%. Compared with compared\_model\_2, DP1 for the proposed method is approximately 25% higher. For DP2, the detection performance of the proposed method is approximately 97%, which is very close to the optimal model performance, as shown in Fig. 10.

## V. CONCLUSION

In this paper, we proposed a new RNN-based learning method that dynamically performs optimal duty cycle operation mode control for individual sensor nodes to monitor and track moving objects in WSNs. Determination of the optimal duty cycle aims to maximize energy efficiency, while satisfying the requirements for object detection. To derive the optimal duty cycle solution in the given environment for RNN supervised learning, we proposed an optimal solution derivation method based on a simulation model in the digital twin space. The current environment was expressed in the form of the position of the sensor nodes and the object detection result of each sensor node in sequential time steps. We presented a structure for LSTM-based RNN learning using the training data obtained in the digital twin space. We evaluated the performance of the proposed RNN-based model in terms of the optimal mode decision accuracy, energy efficiency, and object detection requirement satisfaction through simulations of various object movement conditions. The performance of the proposed model was compared with that of an ideal optimal model and two conventional models using fixed duty cycles. In the evaluation of mode decision accuracy from two perspectives, the proposed method showed an accuracy of 81–98%. In the comparison of energy efficiency, the proposed method showed a performance improvement of 50–70% for the single-object case, and 30–50% for the multi-object case, compared to the methods using the existing fixed duty cycle modes. In addition, in the evaluation of the satisfaction ratio

for the object detection requirements, the proposed method showed 93–97% compliance performance. We demonstrated that the dynamic duty cycle control method proposed in this paper satisfies the object detection requirements at a high level, with minimal information in various object movements and sensor placement environments, while achieving energy efficiency close to the theoretical optimal performance.

## REFERENCES

- [1] A. Iqbal, F. Ullah, H. Anwar, A. Ur Rehman, K. Shah, A. Baig, S. Ali, S. J. Yoo, and K. S. Kwak, “Wearable Internet-of-Things platform for human activity recognition and health care,” *Int. J. Distrib. Sensor Netw.*, vol. 16, no. 6, pp. 1–13, 2020.
- [2] A. Adeel, M. Gogate, S. Farooq, C. Ieracitano, K. Dashtipour, H. Larjani, and A. Hussain, “A survey on the role of wireless sensor networks and IoT in disaster management,” in *Geological Disaster Monitoring Based on Sensor Networks*. Singapore: Springer, 2018, pp. 57–66.
- [3] S. Balaji, K. Nathani, and R. Santhakumar, “IoT technology, applications and challenges: A contemporary survey,” *Wireless Pers. Commun.*, vol. 108, no. 1, pp. 363–388, 2019.
- [4] A. A. Batabyal and H. Beladi, “The optimal provision of information and communication technologies in smart cities,” *Technol. Forecasting Social Change*, vol. 147, pp. 216–220, Oct. 2019.
- [5] A. Abid, F. Khan, M. Hayat, and W. Khan, “Real-time object tracking in wireless sensor network,” in *Proc. 10th Int. Conf. Elect. Electron. Eng. (ELECO)*, Bursa, Turkey, Nov./Dec. 2017, pp. 1103–1107.
- [6] M. S. Adam, M. H. Anisi, and I. Ali, “Object tracking sensor networks in smart cities: Taxonomy, architecture, applications, research challenges and future directions,” *Future Gener. Comput. Syst.*, vol. 107, pp. 909–923, Jun. 2020.
- [7] H. Yetgin, K. T. K. Cheung, M. El-Hajjar, and L. H. Hanzo, “A survey of network lifetime maximization techniques in wireless sensor networks,” *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 828–854, 2nd Quart., 2017.
- [8] Y. Jin, K. S. Kwak, and S.-J. Yoo, “A novel energy supply strategy for stable sensor data delivery in wireless sensor networks,” *IEEE Syst. J.*, vol. 14, no. 3, pp. 3418–3429, Sep. 2020.
- [9] J. Chen, K. Cao, K. Li, and Y. Sun, “Distributed sensor activation algorithm for target tracking with binary sensor networks,” *Cluster Comput.*, vol. 14, no. 1, pp. 55–64, 2011.
- [10] B. Niu, H. Qi, K. Li, X. Liu, and W. Xue, “Dynamic scheming the duty cycle in the opportunistic routing sensor network,” *Concurrency Comput., Pract. Exper.*, vol. 29, no. 17, pp. 1–14, 2017.
- [11] Y. Liu, A. Liu, N. Zhang, X. Liu, M. Ma, and Y. Hu, “DDC: Dynamic duty cycle for improving delay and energy efficiency in wireless sensor networks,” *J. Netw. Comput. Appl.*, vol. 131, pp. 16–27, Jan. 2019.
- [12] C. Jiang, T.-S. Li, J.-B. Liang, and H. Wu, “Low-latency and energy-efficient data preservation mechanism in low-duty-cycle sensor networks,” *Sensors*, vol. 17, no. 5, p. 1051, May 2017.
- [13] S. Sha, J. Li, K. Zhang, Z. Yang, Z. Wei, X. Li, and X. Zhu, “RNN-based subway passenger flow rolling prediction,” *IEEE Access*, vol. 8, pp. 15232–15240, 2020.
- [14] K. Min, D. Kim, J. Park, and K. Huh, “RNN-based path prediction of obstacle vehicles with deep ensemble,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 10, pp. 10252–10256, Oct. 2019.
- [15] J. Luo, Z. Zhang, C. Liu, and H. Luo, “Reliable and cooperative target tracking based on WSN and WiFi in indoor wireless networks,” *IEEE Access*, vol. 6, pp. 24846–24855, 2018.
- [16] L. Liu, G. Han, J. Shen, W. Zhang, and Y. Liu, “Diffusion distance-based predictive tracking for continuous objects in industrial wireless sensor networks,” *Mobile Netw. Appl.*, vol. 24, no. 3, pp. 971–982, Jun. 2019.
- [17] A. H. Mohajerzadeh, H. Jahedinia, Z. Izadi-Ghodousi, D. Abbasinezhad-Mood, and M. Salehi, “Efficient target tracking in directional sensor networks with selective target area’s coverage,” *Telecommun. Syst.*, vol. 68, no. 1, pp. 47–65, 2018.
- [18] J. Zhang, L. Xu, S. Zhou, and X. Ye, “A novel sleep scheduling scheme in green wireless sensor networks,” *J. Supercomput.*, vol. 71, no. 3, pp. 1067–1094, Mar. 2015.
- [19] B. Jiang, B. Ravindran, and H. Cho, “Probability-based prediction and sleep scheduling for energy-efficient target tracking in sensor networks,” *IEEE Trans. Mobile Comput.*, vol. 12, no. 4, pp. 735–747, Apr. 2013.

- [20] P. Medagliani, J. Leguay, G. Ferrari, V. Gay, and M. Lopez-Ramos, "Energy-efficient mobile target detection in wireless sensor networks with random node deployment and partial coverage," *Pervasive Mobile Comput.*, vol. 8, no. 3, pp. 429–447, 2012.
- [21] T. Kim, H. Min, and J. Jung, "A mobility-aware adaptive duty cycling mechanism for tracking objects during tunnel excavation," *Sensors*, vol. 17, no. 3, p. 435, Feb. 2017.
- [22] J. Prajapati and S. C. Jain, "Machine learning techniques and challenges in wireless sensor networks," in *Proc. 2nd Int. Conf. Inventive Commun. Comput. Technol. (ICICCT)*, Coimbatore, India, Apr. 2018, pp. 233–238.
- [23] D. P. Kumar, A. Tarachand, and C. S. R. Annavarapu, "Machine learning algorithms for wireless sensor networks: A survey," *Inf. Fusion*, vol. 49, pp. 1–25, Sep. 2019.
- [24] H. J. Na and S. Yoo, "PSO-based dynamic UAV positioning algorithm for sensing information acquisition in wireless sensor networks," *IEEE Access*, vol. 7, pp. 77499–77513, Jun. 2019.
- [25] W.-K. Yun and S.-J. Yoo, "Q-learning-based data-aggregation-aware energy-efficient routing protocol for wireless sensor networks," *IEEE Access*, vol. 9, pp. 10737–10750, 2021.
- [26] X. Wei, Y. Liu, S. Gao, X. Wang, and H. Yue, "An RNN-based delay-guaranteed monitoring framework in underwater wireless sensor networks," *IEEE Access*, vol. 7, pp. 25959–25971, 2019.
- [27] S. N. Mohanty, E. L. Lydia, M. Elhoseny, M. M. G. Al Otaibi, and K. Shankar, "Deep learning with LSTM based distributed data mining model for energy efficient wireless sensor networks," *Phys. Commun.*, vol. 40, Jun. 2020, Art. no. 101097.
- [28] A. H. Sodhro, S. Pirbhulal, G. H. Sodhro, A. Gurtov, M. Muzammal, and Z. Luo, "A joint transmission power control and duty-cycle approach for smart healthcare system," *IEEE Sensors J.*, vol. 19, no. 19, pp. 8479–8486, Oct. 2019.
- [29] Y. Yu, X. Si, C. Hu, and Z. Jianxun, "A review of recurrent neural networks: LSTM cells and network architectures," *Neural Comput.*, vol. 31, no. 7, pp. 1235–1270, 2019.
- [30] K. Park, Y. Choi, W. J. Choi, H.-Y. Ryu, and H. Kim, "LSTM-based battery remaining useful life prediction with multi-channel charging profiles," *IEEE Access*, vol. 8, pp. 20786–20798, 2020.
- [31] P. M. Djuric, M. Vemula, and M. F. Bugallo, "Target tracking by particle filtering in binary sensor networks," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2229–2238, Jun. 2008.
- [32] J. Yoon, M. Liu, and B. Noble, "Random waypoint considered harmful," in *Proc. IEEE 22nd Annu. Joint Conf. IEEE Comput. Commun. Societies (INFOCOM)*, San Francisco, CA, USA, Mar./Apr. 2003, pp. 1312–1321.



**SEUNG-HEE CHOI** received the B.S. degree from the Information and Communication Engineering Department, Inha University, South Korea, where she is currently pursuing the M.S. degree with the Multimedia Network Laboratory. Her research interests include wireless sensor networks, the IoT, machine learning, deep learning, and reinforcement learning.



**SANG-JO YOO** (Member, IEEE) received the B.S. degree in electronic communication engineering from Hanyang University, Seoul, South Korea, in 1988, and the M.S. and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology, in 1990 and 2000, respectively. From 1990 to 2001, he was a Member of the Technical Staff with the KT Research and Development Group, where he was involved in communication protocol conformance testing and network design fields. From 1994 to 1995 and from 2007 to 2008, he was a Guest Researcher with the National Institute of Standards and Technology, USA. Since 2001, he has been with Inha University, where he is currently a Professor with the Information and Communication Engineering Department. His current research interests include cognitive radio network protocols, VANET and FANET protocol design, AI-based networking algorithms, and the IoT applications.

• • •