applied sciences

MDPI

Article

A Decision Tree-Based Pattern Classification and Regression for a Mobility Support Scheme in Industrial Wireless Sensor Networks

Cheonyong Kim [1] and Sangdae Kim [2,*]

[1] Department of Computer Software, Korea National University of Transportation, Chungju 27469, Republic of Korea; cykim@ut.ac.kr
[2] Department of Medical IT Engineering, College of Software Convergence, Soonchunhyang University, Asan 31538, Republic of Korea
* Correspondence: sdkim.mie@sch.ac.kr

Abstract: Industrial wireless sensor networks (IWSNs) are exploited to achieve various purposes, including enhancing productivity and reducing cost in a variety of industries, and they require low-delay and high-reliability packet transmission. To achieve these requirements, a network manager is responsible for constructing a graph, allocating resources, and determining the transmission cycle and path of each node in advance. However, this approach is inadequate for exploiting mobile devices that constantly change network topology because frequent graph reconstruction and resource reallocation are required. In other words, despite the increasing reliance on mobile devices in a variety of industries, existing schemes cannot adequately respond to path failures due to device movement and subsequent packet loss during recovery. For example, real-time tracking of mobile vehicles in mining operations is crucial for safety and efficiency, where path failures and packet loss can lead to significant issues. To solve this problem, we propose a mobility support scheme to prevent packet loss caused by device mobility. In the proposed scheme, we first classify mobility patterns based on the decision tree and then apply regression to predict their trajectories. By leveraging this predictive information, the network manager could preemptively adjust graph construction and resource allocation to accommodate topology changes. Performance evaluation results show that the predicted mobility patterns closely match the actual patterns, achieving a high packet delivery ratio compared to conventional schemes, while also enabling efficient resource management.

Keywords: industrial wireless sensor networks (IWSNs); decision tree; regression; mobility pattern prediction; graph construction; resource allocation

Academic Editors: Mahdi Zareei and Jesús Arturo Pérez-Díaz

Received: 31 December 2024
Revised: 27 January 2025
Accepted: 28 January 2025
Published: 30 January 2025

Citation: Kim, C.; Kim, S. A Decision Tree-Based Pattern Classification and Regression for a Mobility Support Scheme in Industrial Wireless Sensor Networks. *Appl. Sci.* **2025**, *15*, 1408. https://doi.org/10.3390/app15031408

1. Introduction

The industrial wireless sensor networks represented by WirelessHART [1] are a type of industrial Internet and are used in a variety of industries, including manufacturing, mining, and agriculture [2]. This network collects information on various devices and environments used in each field and makes decisions that increase profits, improve efficiency and productivity, and reduce costs in each industry [3,4]. For example, in the case of a monitoring system [5], by continuously monitoring the condition of the current operating equipment, damage and failure of the equipment can be predicted in advance to prevent quality or productivity deterioration due to equipment faults. In the case of an inventory management system [6], it is possible to identify the amount of raw materials and product loading needed to prevent shortages of raw materials and overloading of products. In addition, cost savings can be achieved by automating processes using information sensed by nearby sensors.

To utilize IWSNs in various fields, as described above, data collection based on low delays and a high reliability of data transmission is required [7,8]. However, in contrast to traditional wired networks [9,10], wireless networks are vulnerable to collisions and interference between transmissions [11]. Therefore, to address collision and interference issues, IWSNs perform graph construction and resource allocation through a network manager. That is, the network manager collects information about all of the devices that comprise the network to construct a directed graph for the entire network topology. Thereafter, resource allocation is performed using time-slotted channel hopping (TSCH) based on the constructed graph. This determines at what time each device delivers data to which device and via which channel. This process solves the problem of collisions and interference between data transmissions among devices.

However, IWSNs have been designed with a focus on alleviating the disadvantages of wired systems, such as difficulties in installation and wiring and facility protection [12]. Therefore, they do not adequately respond to topology changes caused by the movement of workers and mobile devices [13,14]. For example, when collecting worker information to ensure worker safety, such as in the FASyS project, network topology changes will occur due to the movement of workers. However, it is difficult for existing schemes to cope with network topology changes. In other words, the graph construction and resource allocation needed to cope with the topology changes have to be performed again, and communication between the mobile devices and the network are lost during this process. In addition, as devices that deliver packets through mobile devices are also affected, the overall network performance is degraded, which could cause fatal problems in various industrial applications.

Therefore, we propose a mobility support scheme based on decision tree-based pattern classification and regression. First, the decision tree is exploited to classify the diverse mobility patterns of mobile devices. By analyzing mobility data, the decision tree categorizes movement patterns into linear, cyclic, or random movements. The proposed scheme utilizes a decision tree for mobility pattern classification due to its interpretability, computational efficiency, and suitability for real-time applications in industrial wireless sensor networks (IWSNs). Unlike more complex models, such as neural networks, decision trees provide clear and interpretable decision boundaries, which are crucial for resource allocation in dynamic industrial environments. While neural networks may offer higher accuracy in some scenarios, their reliance on large datasets and high computational requirements makes them less practical for resource-constrained environments like IWSNs. Once the mobility pattern is identified, each pattern is assigned to the most suitable predictive model. For example, devices classified as moving along relatively regular or polynomial trajectories employ regression to estimate the mobility pattern. On the other hand, if the device exhibits highly complex or non-linear movements that make accurate prediction extremely difficult, the existing resource allocation scheme is employed [15]. By combining decision tree-based classification with pattern-specific regression, the proposed scheme could enable more accurate mobility pattern predictions and reduce packet loss caused by topology changes.

To exploit these predictions effectively, we perform graph construction and resource allocation for communication with mobile devices on all devices based on [15]. Through the pre-resource allocation scheme, when a mobile device transmits data to the network manager via a nearby device, the network manager could collect the location information of the mobile devices and the exact transmission time. Based on the information, the network manager constructs a routing graph and allocates resources based on the time-specific network topology predicted by the decision tree-based classification and the regression. This scheme ensures that the network can proactively adapt to network changes, thus providing higher reliability and managing resource efficiently in industrial wireless sensor network environments.

The key contributions of this paper are as follows:

- Our scheme enhances the accuracy of mobility pattern classification to overcome the limitations of the traditional scheme, which relies on a single regression/prediction model to cover all mobility patterns.
- The proposed scheme improves prediction accuracy and computational efficiency by employing customized predictive models tailored to each classified type.
- Based on these advantages, the proposed scheme enables an improved transmission success ratio and more efficient resource allocation.

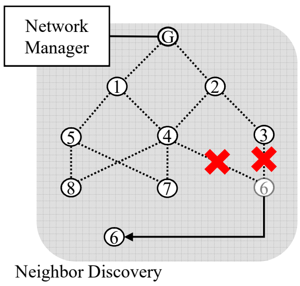
The remainder of this paper is organized as follows. In Section 2, we explain the operating process of industrial wireless sensor networks represented by WirelessHART and the problems caused by the movement of devices in the existing network. In Section 3, we explain a mobility support scheme based on a decision tree-based pattern classification and regression to solve the problems, and performance evaluation is described in Section 4. Finally, we summarize this paper and outline future works in Section 5.

2. Related Works and Problem Statement

In this section, we provide an overview and operation of WirelessHART, a typical example of an industrial wireless sensor network, and define the problems caused by the movement of mobile devices. In addition, we discuss related works to solving this problem.

2.1. An Overview and Operation of WirelessHART

In this section, we describe the overview and operation process of WirelessHART. Figure 1 shows an example of a WirelessHART topology and its resource allocation.



(1 TS = 10ms)

TS / CH.	0	1	2	3	4	5	...
0	8→5	5→1	1→GW				
1		8→7	7→4	4→1	1→GW		
2	6→3	3→2		2→GW			
3		6→4				Discovery	
...							

(The discovery slot is used by nodes which discovery timer has expired to send discovery messages.)

Figure 1. An example of WirelessHART network and resource allocation.

The network consists of field devices, a gateway, and a network manager. The field devices are physically deployed sensors that sense information from a device or environment and transmit the sensed data to the gateway. The gateway performs the role of connecting the WirelessHART network with the external backbone network. It is the endpoint of the data collected from each field device.

The network manager configures and manages networks for low-delay and high-reliable data transmission required by industrial environments. To achieve this goal, the manager constructs a graph of the network topology and allocates resources based on device information collected through the gateway. Especially, by exploiting TSCH in the resource allocation process, transmission collisions and interference that could occur in the data transmission process of each device are prevented. In addition, it adjusts the transmission path and time when network changes such as addition or removal of devices occur. That is, the network manager performs the core functions of the WirelessHART network and is specifically aimed at responding to changes in the network. Typically, there is the join process to involve newly added devices in the network and the neighbor discovery process to deal with topology changes caused by device location changes and environmental changes.

The neighbor discovery process [16–18] is exploited to address changes in the location of devices which have already participated in the network through the join process, or when the quality of communication between devices has changed due to environmental changes. To accomplish this process, field devices transmit discovery messages for each period determined by the network manager, which functions similarly to a keep-alive message while also searching for new paths. In other words, the discovery message is exploited to periodically check whether the neighborhood near it operates normally when there is no change in the network topology. However, if the network changes occur, some field devices receive a discovery message from a new field device or fail to receive a discovery message from the existing field devices. In this case, the field devices report the detected changes to the network manager. The network manager, which receives the report, performs graph reconstruction and resource reallocation according to the changed network topology. Finally, it informs each field device of the information recalculated for communication in the changed topology.

2.2. Problem Definition Caused by Movement of Mobile Devices

As mentioned above in Section 2.1, WirelessHART exploits neighbor discovery to respond to network changes. However, because discovery messages are sent at intervals predetermined by the network manager, it is difficult to respond quickly to network changes. That is, the existing discovery process has difficulty responding to continuous topology changes caused by device movement.

For example, let us assume that device 6 is a mobile device in Figure 1 and is connected to the network through the join process. The network manager constructs a graph for communication for device 6 and allocates resources. After resource allocation, even if device 6 moves close to device 7 and device 8 and disconnects from device 3 and device 4, this disconnection cannot be resolved until the network manager recognizes the change in the network topology through the neighbor discovery process. In other words, until the network changes are discovered by the neighbor discovery process, device 6 would attempt to transmit its sensed data to the gateway through device 3 or device 4 at a designated time and channel, but all data would be lost. This problem is not only a problem for mobile devices but also extends its adverse effects to child nodes with mobile devices as parents. Additionally, even if the network manager constructs a new graph and reallocates resources for network changes discovered through the neighbor discovery process, the discovered location may differ from the actual current location due to device mobility. In conclusion, network changes caused by the movement of devices could lead to communication instability and disconnections in the existing WirelessHART network.

2.3. Related Works

In this section, we review related studies that address the challenges caused by the mobility of devices in industrial wireless sensor networks (IWSNs), focusing on neighbor discovery, mobility prediction, and resource allocation strategies.

Table 1 summarizes the key approaches, their strengths, and limitations discussed in the Related Works section. These insights highlight the gaps in existing studies that our proposed scheme addresses. Montero et al. [16] proposed the LAN-ND (Listen Advertise Network Neighbour Discovery) protocol, which enhances WirelessHART's neighbor detection capabilities by leveraging deterministic scheduling of advertisement links. While effective in improving detection speed and reliability, the protocol faces challenges in energy efficiency under highly dynamic conditions. Montero et al. [18] introduced neighbor discovery protocols for mobile WirelessHART networks, emphasizing the integration of deterministic and probabilistic methods to reduce latency and improve reliability. Their findings highlight the need for efficient discovery mechanisms in dynamic industrial environments with mobile nodes. Kim et al. [19] introduced a machine learning-based mobility support scheme for IWSNs. This approach demonstrated improvements in packet delivery ratio and resource efficiency compared to traditional methods but primarily focused on linear regression and did not fully address non-linear or random mobility patterns. Alhulayil et al. [20] investigated resource allocation strategies in dual-hop cognitive relay networks with amplify-and-forward relays, demonstrating the benefits of best-relay selection (BRS) under spectrum-sharing constraints. Their findings highlight the importance of efficient resource utilization in dynamic environments, which is a significant consideration for industrial networks.

Table 1. Summary of related works.

Study	Strengths	Limitations
Montero et al. (2013) [16]	Improves neighbor detection speed and reliability in dynamic environments	Energy efficiency is limited in highly dynamic scenarios
Montero et al. (2017) [18]	Reduces latency and improves reliability in mobile networks	Scalability and energy consumption in larger networks are not fully addressed
Kim et al. (2020) [19]	Enhances packet delivery ratio and resource efficiency	Focuses primarily on linear patterns; does not address non-linear or random mobility patterns
Alhulayil et al. (2023) [20]	Improves resource utilization and network performance in dynamic environments	Applicability to large-scale industrial networks with mobility is not explored

These studies collectively emphasize the importance of efficient neighbor discovery, mobility prediction, and resource allocation in dynamic IWSN environments. Building on these insights, our proposed scheme leverages decision tree-based classification and regression models to dynamically adapt to diverse mobility patterns. Unlike existing approaches, our proposed scheme improves prediction accuracy, enhances resource efficiency, and ensures reliability in industrial settings with complex mobility dynamics.

3. Mobility Support Scheme Based on Decision Tree-Based Pattern Classification and Regression

In this section, we propose a mobility support scheme that identifies changes in the location of devices with mobility in advance and exploits this information for network operation. The proposed scheme consists of two phases: (1) mobility pattern classification based on a decision tree and (2) mobility path prediction based on regression. Through this proposed scheme, frequent topology changes in industrial wireless sensor networks could be efficiently managed. Through the decision tree, the mobility trajectory is identified as being closer to one of the following patterns: linear, non-linear, or random. Based on the classification results, a customized prediction model (e.g., linear regression, polynomial regression) is applied to predict future locations or paths. The proposed scheme could enhance network performance by minimizing packet loss and managing resources efficiently.

3.1. Decision Tree-Based Mobility Pattern Classification

In this section, we provide a detailed explanation of the decision tree-based mobility pattern classification process, which classifies mobility trajectories into patterns (linear, non-linear, random) using mobility trajectory data. This process represents the first step of the mobility support scheme and is a key process related to the selection of subsequent prediction models.

To prepare data for training the decision tree, the location data of each mobile device is collected at regular time intervals in the form of (t,x,y). Multiple trajectories are gathered, and each is labeled as linear, non-linear, or random to construct the training datasets. The Figure 2 shows an example of coordinates according to the mobile device.

Time (s)	0	10	20	30	40	50	60	70	80	90	...
Position (coordinates)	(0, 0)	(0, 10)	(0, 20)	(0, 30)	(0, 40)	(0, 50)	(0, 60)	(0, 70)	(0, 80)	(0, 90)	...

(a) Example of coordinates for a device moving linearly

Time (s)	0	10	20	30	40	50	60	70	80	90	...
Position (coordinates)	(0, 0)	(0, 10)	(0, 20)	(0, 20)	(0, 10)	(0, 0)	(0, 10)	(0, 20)	(0, 10)	(0, 0)	...

(b) Example of coordinates for a device moving non-linearly

Time (s)	0	10	20	30	40	50	60	70	80	90	...
Position (coordinates)	(0, 0)	(10, -10)	(-10, -30)	(-10, -40)	(-30, -50)	(-40, -30)	(-50, -20)	(-40, -10)	(-30, 0)	(-20, 10)	...

(c) Example of coordinates for a device moving randomly

Figure 2. An example of the training set for learning locations by unit time of mobile device.

After collecting and labeling the mobility trajectories (linear, non-linear, or random), we exploit two primary features from each trajectory to learn the decision tree: the coefficient of determination (R^2) and the standard deviation of direction changes ($dangle\_std$). The R^2 value is obtained by fitting a simple linear regression model with respect to time, thereby indicating how closely the trajectory aligns with a straight line. The $dangle\_std$ is derived by computing the angle $\theta = arctan2(\Delta y, \Delta x)$ for consecutive positions and measuring the variability ($\Delta\theta$) across the entire path. As a result, trajectories with high R^2 and low $dangle\_std$ are likely to be classified as linear, whereas irregular direction changes and low linearity would be identified as non-linear or random.

Figure 3 shows the example of a learned decision tree, where the root node first checks whether $R^2 \leq 0.575$. If R^2 exceeds this threshold, the trajectory would be classified as "Linear", meaning that the trajectory is suitable for the linear regression model. In contrast, if R^2 remains below 0.575, the model proceeds to evaluate $dangle\_std$. Lower $dangle\_std$ values, such as those under 2.596, indicate0"Non-linear", meaning that the trajectory is suitable for the polynomial regression model. However, when $dangle\_std$ surpasses a certain threshold, the trajectory would be classified as "Random". It indicates highly variable direction changes. In this way, each node of the decision tree refines how the trajectory is classified by considering R^2 against $dangle\_std$. As a result, samples with high R^2 and low $dangle_{s}td$ are directed to the "Linear" category, moderately curved paths are classified into the "Non-linear" category, and erratic mobility is categorized into the "Random" category.

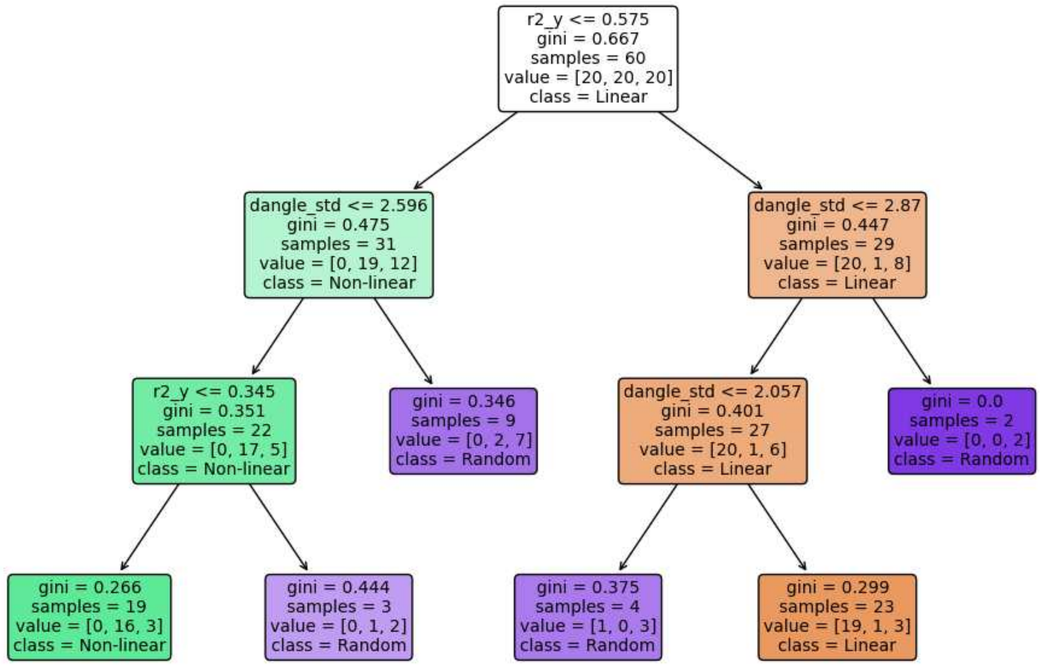


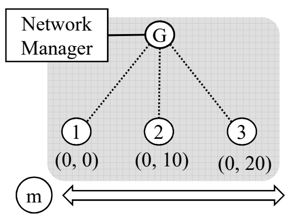
Figure 3. An example of learned decision tree.

Each threshold used in the decision tree play a crucial role in accurately classifying the mobility patterns. R^2, which serves as a key metric for evaluating the linearity of movement paths, was selected by the decision tree during the learning process to achieve optimal classification accuracy. Similarly, *dangle_std*, which measures the variability of angular changes, was set to effectively differentiate between non-linear and random paths. Since these thresholds were determined during the learning process, they avoid overfitting to specific datasets and could be reliably applied to various scenarios in industrial environments.

3.2. Regression-Based Mobility Pattern Prediction

In this section, we propose a scheme for predicting mobility patterns based on the classes determined by the decision tree. For relatively simple linear trajectories (classified as "Linear"), a simple approach is sufficient for prediction and does not require a complex scheme. Therefore, the focus is on explaining polynomial regression for non-linear mobility patterns.

The trajectory corresponding to the Figure 2b could be observed as shown in Figure 4, where each coordinate could be matched with its respective device identification number. Each field device senses a device or environment according to a predetermined role and periodically transmits this data to the gateway. In the above training set, input X is the unit time when the mobile device transmits data through the neighbor device, and output Y is a label (device identification number) for the neighbor device that has received the data from the mobile device.



Time (s)	0	10	20	30	40	...
Position (coordinates)	(0, 0)	(0, 10)	(0, 20)	(0, 20)	(0, 10)	...
Position (node number)	1	2	3	2	1	...

Figure 4. An example of a simple topology with a mobile device and converted training data sets.

Figure 5 is a schematic diagram of the training data for input X and output Y, which is trained using polynomial regression based on Gaussian basis functions. The prediction function to which m Gaussian basis functions are applied is expressed as Equation (1) below.

$$y(x, \mathbf{w}) = w_0\phi_0(x) + w_1\phi_1(x) + w_2\phi_2(x) + \ldots + w_n\phi_n(x) \quad (1)$$

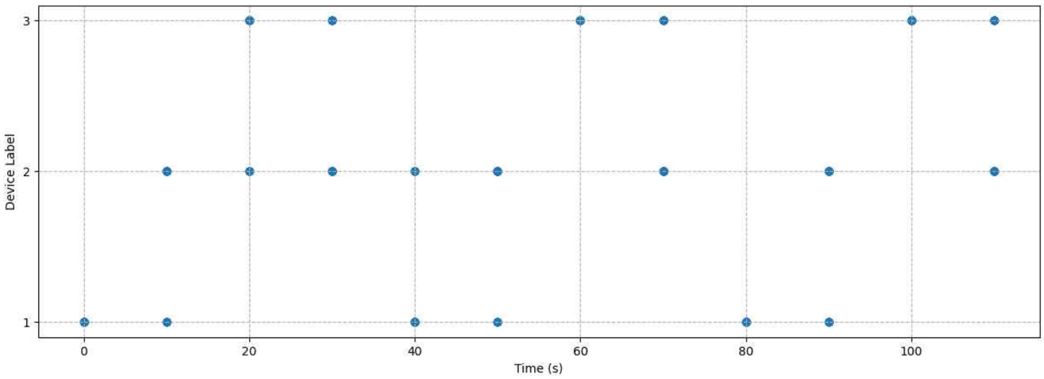


Figure 5. A schematized training data sets.

This prediction function should be optimized to yield a parameter w with the optimal mean square error. For this purpose, we first need to determine the number of basis functions m. To determine the number of basis functions m, we exploit the K-fold cross-validation scheme by increasing m. K-fold cross-validation splits the training set into K parts and exploits one of them as the test set and the rest as the training set. We use the training set to optimize the parameters of the model and calculate the mean squared error on the test set, repeating this process for each number of basis functions m to evaluate the mean square error and determine the optimal number of basis functions.

Figure 6 presents the graph of the fitting results for the number of basis functions, with m being 5, 7, 9, and 11, respectively, as determined by performing K-fold cross-validation. As the number of basis functions increases, the complexity of the function increases, and it might be observed that the prediction model fits closer to the training set. Accordingly, it might be confirmed that the error between the training and test sets gradually decreases. However, when the number of basis functions reaches a certain level, it can be seen that the error is no longer reduced. Therefore, we utilize nine basis functions as a heuristic approach based on a given training set. The location of the mobile device at each unit of time through the predictive model is as follows.

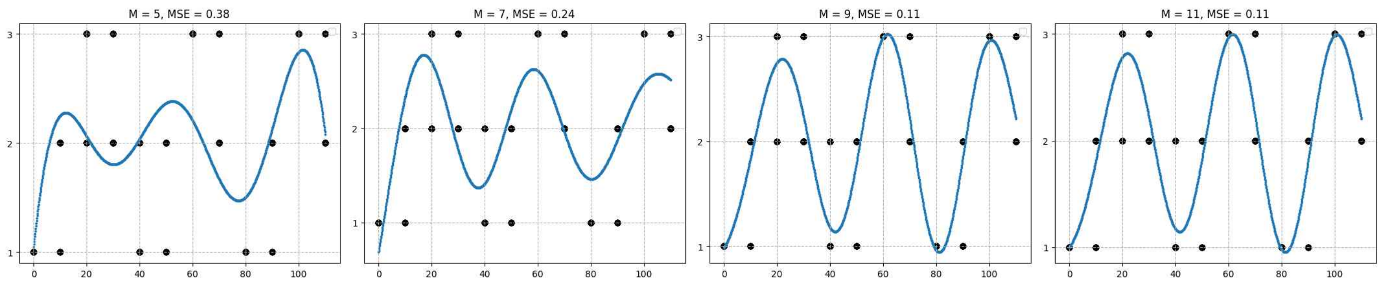


Figure 6. The fitting results according to number of basis functions m.

Figure 7 shows the predicted locations of the mobile devices identified based on the prediction model trained on the data in Figure 4. By referring to Figures 4 and 7, it could be explained that at cycle 0, the mobile device transmitted data through either device 0 or 1. The prediction result of 0.97 indicates that data were mostly sent through device

1. Similarly, on cycle 10, the mobile device transmitted data through either device 1 or 2. The prediction result of 1.86 indicates that data were mostly sent through device 2. To determine the device to perform the communication, we rounded the training results to identify the device. As a result, comparing the actual movements of the mobile devices with the predicted model indicates an accuracy of about 86%, reflecting a relatively high level of prediction for the actual movements of the mobile devices. However, while a prediction accuracy of 86% is not low, it might not be sufficient for real-world industrial scenarios. To address this, the proposed scheme allocates resources not only to the location predicted by the regression model but also to devices located at the predicted device's previous and subsequent positions in time. (See Section 3.3 for more information).

X (input)	0	10	20	30	40	50	60	70	80	90	100	110
Y (predicted device label)	0.97	1.86	2.72	2.25	1.18	1.77	3.02	2.19	0.95	1.86	2.98	2.22

Figure 7. The location prediction results for unit time for mobile devices based on predictive models.

When the trajectory is classified as "Random", predicting the location of a mobile device becomes difficult based on polynomial regression as previously described. Alternatively, time-series prediction models such as RNNs could be employed; however, the prediction accuracy for randomly moving devices would significantly decrease. Therefore, in cases where the trajectory is determined to be "Random", we exploit the existing resource allocation scheme for mobility support [15].

3.3. Graph Construction and Resource Allocation Based on Predicted Mobility Pattern

In this section, we describe how to construct a graph for the mobile device and allocate resources according to the mobility pattern predicted in Section 3.1 above. However, the main idea of this paper is to construct a separate graph and allocate each resource according to the mobility pattern and cycle of the mobile device. Therefore, we focus on the graph construction and resource allocation results when the proposed scheme is applied to the scheme of [21], which is one of the representative graph construction and resource allocation schemes in WirelessHART.

Figure 8 shows the examples of the network topology for each cycle and the corresponding uplink graph when the proposed scheme is applied to [21]. First, assume that device 6 in Figure 8 is a mobile device and travels close to devices 1, 2, and 3 as shown in Figure 8a. At this time, Figure 8b shows a graph for communication with device 1, Figure 8c shows a graph for communication with device 2, and Figure 8d shows a graph for communication with device 3. The initial mobile device, device 6, joined the network through a join process near devices 3 and 4, as shown in Figure 8a, and moved along a black solid line at regular intervals. Based on the results learned in Section 3.1 above, the network manager knows that the devices used by the mobile device for communication are device 1, device 2, and device 3, and knows when they are connected to each device. Therefore, at the time of TS 0, TS 400, and TS 800, a graph (Figure 8a) is generated to be used for communication with device 1. At the time of TS 100, TS 300, TS 500, TS 700, TS 900, and TS 1100, a graph (Figure 8c) is constructed to be exploited for communication with device 2. Similarly, at the time of TS 200, TS 600, and TS 1000, a graph (Figure 8d) is constructed to be exploited for communication with device 3. All of the graphs in Figure 8 could be constructed by exploiting [21], but it is difficult to know which graphs to exploit at a certain time. However, by constructing and using graphs according to time-specific access devices based on movement patterns, the proposed scheme avoids path disconnection due to the movement of mobile devices, as well as the need for graph reconstruction and resource reallocation to recover them.

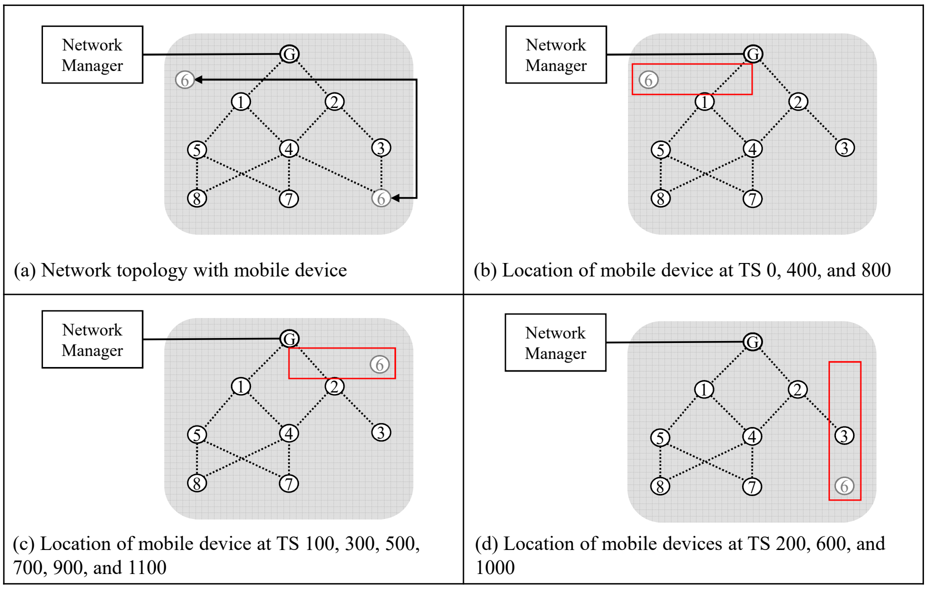


Figure 8. The examples of constructed routing graph based on mobility pattern.

After the graph construction is completed, the network manager allocates resources to each graph. This resource allocation reflects the network topology state according to the mobility pattern of the mobile device. Figures 9–11 are the resource allocation results according to [15,21], and the proposed scheme for Figure 8, respectively. In each allocation table, the vertical axis represents the channel used (CH), while the horizontal axis represents time (TS). Each cell in the table, denoted as X→Y, indicates that data are transmitted from device X to device Y. For instance, TS 0/CH. 0 in Figure 9 indicates that device 8 transmits data to device 7 via channel 1 at time slot TS 0. The shaded portion of each table represents the section assigned by the mobile device for data transmission.

Based on this, Figure 9 illustrates the resource allocation when mobile device 6 connects to the network via device 3 and device 4 through the join process. In the existing scheme [21], as shown in Figure 9, resource allocation is performed based on the current position without accounting for the movement of the mobile device. As illustrated in Figure 9, resource allocation is repeated at the time of generating the graph, irrespective of the movement of the mobile device. Consequently, at TS 100 and TS 200, the mobile device cannot transfer data through the allocated resources.

In the case of Figure 10, as a basic scheme to support the mobility of mobile devices, resources would be pre-allocated to devices at all locations where the mobile device is likely to be. This allows data to be transmitted through surrounding devices whenever and wherever the mobile device is present. However, it can be observed that a significant amount of resources are wasted in supporting the mobile device's communication when compared to Figure 9 or Figure 11. This waste occurs because resources are allocated to fixed devices independently of the actual location of the mobile device.

Figure 11 shows that the proposed scheme allocates resources based on the movement patterns of mobile devices, allowing mobile device 6 to communicate with device 1 at TS 0, device 2 at TS 100, and device 3 at TS 200, as predicted in Section 3.1. Additionally, to prevent transmission failures due to prediction errors in the predictive model, additional resource allocation is performed at specific time slots, such as TS 1, TS 101, TS 102, and TS 201. Through this resource allocation process, it is possible to prevent transmission failures due to the movement of mobile devices, which could occur when resources are allocated, as shown in Figure 9. Additionally, the proposed scheme addresses the inefficiency problem of excessive resource use for mobility support, as depicted in Figure 10.

CH. \ TS	0	1	2	3	4	...	100	101	102	...	200	201	202	...
0	8→7	6→3	6→4	4→2	3→4		8→7	6→3	6→4		8→7	6→3	6→4	
1		8→5	3→2	5→7	2→GW			8→5	3→2			8→5	3→2	
2			5→1						5→1				5→1	
3														
...														

Figure 9. An example of resource allocation by the existing scheme [21].

CH. \ TS	0	1	2	3	4	...	100	101	102	...	200	201	202	...
0	6→8	6→3	6→7	6→4	6→2		6→8	6→3	6→7		6→8	6→3	6→7	
1		8→7	8→5	5→7	7→4			8→7	8→5			8→7	8→5	
2			3→4	3→2	5→1				3→4				3→4	
3														
...														

Figure 10. An example of resource allocation by the existing scheme [15].

CH. \ TS	0	1	2	3	4	...	100	101	102	...	200	201	202	...
0	8→7	8→5	5→7	7→4	4→1		8→7	8→5	5→7		8→7	8→5	5→7	
1	6→1	3→4	3→2	5→1			6→2	3→4			6→3	6→4	3→2	
2		6→2						6→1	6→3			6→2		
3														
...														

Figure 11. An example of resource allocation by the proposed scheme.

4. Performance Evaluation

In this section, we describe the performance evaluation results of the transmission success ratio and resource occupancy ratio. We compare scenarios where the mobile device moves between networks using the graph construction and resource allocation schemes from [15,21], and the proposed schemes. First, in Section 4.1, we describe the performance evaluation environment and factors. In Section 4.2, we evaluate the classification performance of the decision tree for linear, non-linear, and random patterns. We present the performance evaluation results for the average transmission success ratio based on the device's movement speed in Section 4.3. Section 4.4 presents the results of the performance evaluation and analysis of the average transmission success ratio based on the number of mobile devices. Finally, in Section 4.5, we show the resource occupancy ratio results for each scheme.

4.1. Performance Evaluation Environment and Factors

The performance evaluation of both the existing research and the proposed scheme was conducted using the NS-3 network simulator [22]. A network comprising 100 devices arranged in a grid structure within a 100 m $\times$ 100 m area was simulated. In this scenario, five devices for each pattern—linear, non-linear, and random—move at a speed of 10 m/s. To represent different mobility scenarios, the following movement patterns were defined:

Linear patterns: Devices move along straightforward and predictable trajectories, representing scenarios with consistent and uniform mobility.

Non-linear patterns: Devices follow trajectories with moderate variability, introducing complexity to their movement paths while still retaining some level of predictability.

Random patterns: Devices exhibit highly irregular and unpredictable movements, reflecting scenarios with significant uncertainty and complexity in mobility.

Each device has a transmission range of 50 m and transmits data to a gateway at one-second intervals. To evaluate the impact of network changes due to device mobility on transmission success ratio, we assume an ideal transmission success ratio of 100%. Performance evaluations were conducted for 6400 s. The results presented are the average values derived from ten separate simulation runs to ensure reliability and accuracy. The performance evaluation factors and attributes exploited to learn the decision tree are as follows:

The transmission success ratio is defined as the ratio of packets successfully delivered to the gateway to the total number of packets sent by the devices.

The resource occupancy ratio refers to the proportion of resources utilized for network configuration compared to the total available resources.

The R^2 (coefficient of determination) measures the linearity of trajectories. Values closer to 1 indicate a highly linear trajectory, while lower values represent less linear paths.

The *dangle_std* captures the variability of angular changes across a trajectory. A lower value represents smoother and more predictable paths, while a higher value corresponds to more erratic movements.

The dataset used for classifier development consists of a total of 60 samples, distributed equally across three mobility patterns: 20 samples for linear trajectories, 20 for non-linear trajectories, and 20 for random trajectories. Each sample includes the computed values for R^2 and *dangle_std*, which are used as inputs for the decision tree model.

4.2. Confusion Matrix for Mobility Pattern Classification

In this section, we evaluate the accuracy of the classification scheme based on the decision tree using a confusion matrix. The Figure 12 illustrates the confusion matrix obtained when 10 samples each of linear, non-linear, and random trajectories are tested on the trained classifier. This result serves as an initial validation to demonstrate the classifier's behavior under controlled conditions and is not intended to represent a comprehensive evaluation. The horizontal axis represents the predicted labels, while the vertical axis indicates the actual labels. Test results show that the decision tree correctly classified all linear and non-linear trajectories. For random trajectories, nine out of ten were correctly identified, with one misclassified as non-linear. Despite this minor error in classifying random trajectories, the scheme demonstrates strong overall classification performance.

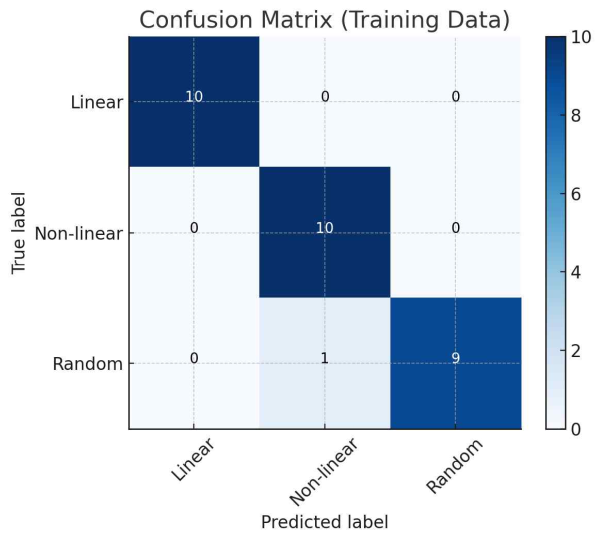


Figure 12. The examples of confusion matrix with initial validation results with 10 samples per class.

4.3. Transmission Success Ratio According to Movement Speed of Mobile Devices

Figure 13 presents the performance evaluation results of the transmission success ratio of all devices relative to the movement speed of the mobile device. The delay in the neighbor discovery process caused by increased mobility speeds leads to more frequent disconnections in the network. In the case of [21], as previously defined in Section 2.2, the movement of mobile devices causes network changes and path disconnection. Usually, the neighbor discovery process is exploited to respond to the problems; however, it does not immediately reflect network changes caused by the movement of mobile devices. Thus, the transmission success ratio tends to decrease. However, In the case of [15], it assumes that the mobile device exists in all communicable locations and allocates all resources in advance. Therefore, it shows the highest transmission success ratio among the comparative protocols. The proposed scheme allocates resources by predicting movement paths for linear and non-linear mobility patterns for each device, while employing the same approach as [15] for random patterns, thereby achieving a high transmission success ratio.

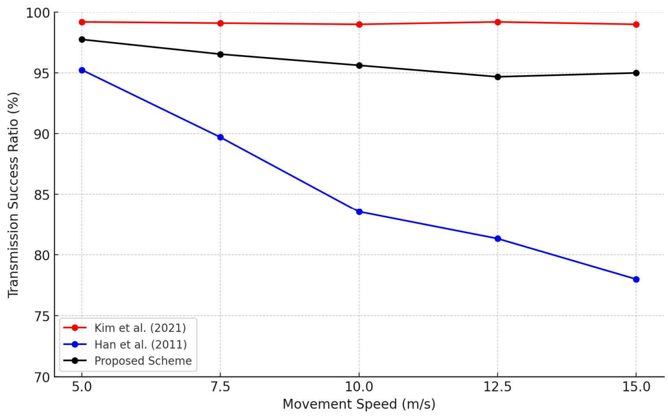


Figure 13. Packet delivery ratio versus movement speed has been analyzed based on [15,21] and proposed scheme.

4.4. Transmission Success Ratio According to the Number of Mobile Devices

Figure 14 presents the performance evaluation results when the number of devices moving in linear, non-linear, and random patterns is increased to 6, 7, 8, 9, and 10, respectively. In the case of [21], similar to the experimental results in Section 4.3, the transmission success ratio decreases as the number of mobile devices increases. This is because, as previously explained, the conventional neighbor discovery process does not respond adequately to mobility. In addition, it hinders the connectivity of devices that should communicate with the gateway through mobile devices. In the case of [21], similar to the experimental results in Section 4.3, a high transmission success ratio is maintained even as the number of mobile devices increases. However, as will be discussed in detail in Section 4.5, this mobility support comes at the cost of highly inefficient resource allocation. Lastly, the proposed scheme also achieves a high transmission success ratio by predicting movement paths and allocating resources based on mobility pattern classification. However, as the number of mobile devices increases, the possibility of errors in pattern classification and prediction also increases, resulting in slightly lower performance compared to [15]. Nevertheless, it uses fewer resources than [15], which allocates a large amount of resources to all mobile devices.

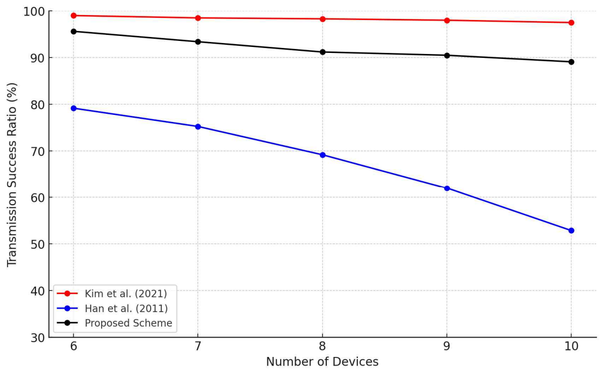


Figure 14. Packet delivery ratio versus number of mobile devices has been analyzed based on [15,21] and proposed scheme.

4.5. Resource Occupancy Ratio

Figure 15 presents the percentage of resources utilized during the operation of each scheme. WirelessHART, a network based on IEEE 802.15.4e, is characterized by limited bandwidth, making efficient resource management a critical factor for scalability. In the case of [15], the resource occupancy ratio is the highest among the compared schemes due to its assumption that all mobile devices could communicate with every fixed device in the network. This scheme ensures network stability through excessive pre-allocation of resources; however, it results in significant resource inefficiency. On the other hand, Ref. [21] achieves the lowest resource occupancy ratio by allocating resources based on a static network topology, without accounting for the mobility of devices. While this scheme minimizes resource usage, it struggles to adapt to dynamic network changes, which might lead to performance degradation in complex environments. In contrast, the proposed scheme classifies mobility patterns—linear, non-linear, and random—and predicts movement paths to allocate resources accordingly. As a result, it shows a slightly higher resource occupancy ratio than [21], while remaining significantly lower than [15]. Additionally, it achieve high transmission success ratio while avoiding unnecessary resource allocation. In other words, the proposed scheme could achieve both scalability and efficient resource management in dynamic network environments. To summarize the results, Ref. [15] achieve high transmission success rates through extensive resource pre-allocation but suffer from high resource occupancy. In contrast, ref. [21] minimizes resource occupancy but becomes less reliable in dynamically changing network environments. The proposed scheme balances between these trade-offs by predicting mobility patterns and allocating resources efficiently. These findings demonstrate that the proposed scheme achieves both a high transmission success rate and superior scalability compared to existing schemes.

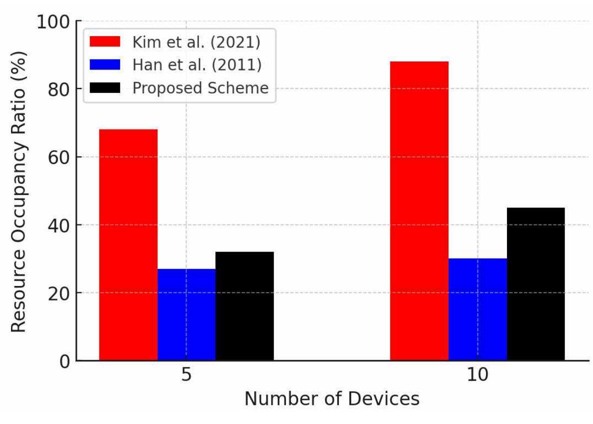


Figure 15. Resource occupancy ratio is calculated based on [15,21] and proposed scheme.

5. Conclusions and Future Works

Industrial wireless sensor networks (IWSNs) exploited in various industries have been designed with a focus on mitigating the drawbacks of wired networks. Therefore, when network topology changes frequently, problems such as increased management costs and performance degradation due to graph reconstruction and resource reallocation occur. Recently, various mobile devices have been exploited in IWSNs, which causes frequent topology changes. Therefore, a strategy is required to address issues like path disconnection and reliability degradation caused by mobile devices during graph reconstruction and resource reallocation. In this paper, we propose a mobility support scheme to prevent packet loss caused by device mobility. The main idea is to employ a decision tree for classifying mobility patterns (linear, non-linear, and random), followed by regression to predict the actual trajectories for each identified pattern. By adopting this scheme, the network manager could proactively detect and respond to network changes, thereby achieving a higher transmission success ratio than traditional methods while also managing resources efficiently. In the performance evaluation results, the proposed scheme achieved high prediction accuracy and a high transmission success ratio even in scenarios where multiple mobile devices continuously changed the network topology. Moreover, by employing prediction methods tailored to each classified mobility pattern, resources can be allocated more efficiently compared to existing schemes, thereby ensuring network scalability.

However, certain limitations remain. Support for devices with random movement patterns remains insufficient. Due to the inherent unpredictability of random mobility, it is very challenging to make accurate forecasts using regression or RNN-type models. Such devices with truly random mobility are expected to increase in future industrial environments, necessitating further research to address this issue.

As part of future research, integrating fuzzy decision tree methodologies, such as those proposed by Zaitseva et al. [23], could provide enhancements. These techniques allow for better handling of uncertainty and variability in attribute importance, offering a more refined evaluation of factors influencing decision-making processes. This approach is expected to improve the prediction of random mobility patterns and optimize resource allocation even under highly dynamic conditions.

Author Contributions: C.K. and S.K. contributed to protocol design and detailed algorithms. They also conducted the performance analyses. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Soonchunhyang University Research Fund (No. 20220441).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. *IEC PAS 62591:2009*; Industrial Communication Networks-Wireless Communication Network and Communication Profiles-Wirelesshart. IEC: Geneva, Switzerland, 2009.
2. Luo, X.; Liu, J.; Zhang, D.; Chang, X. A large-scale web QoS prediction scheme for the Industrial Internet of Things based on a kernel machine learning algorithm. *Comput. Netw.* **2016**, *101*, 81–89. [CrossRef]
3. Sullivan, G.; Pugh, R.; Melendez, A.P.; Hunt, W. *Operations & Maintenance Best Practices—A Guide to Achieving Operational Efficiency (Release 3)*; Technical Report; Pacific Northwest National Lab. (PNNL): Richland, WA, USA, 2010.
4. *Industry 4.0: Huge Potential for Value Creation Waiting to be Tapped*; Deutsche Bank Research: Frankfurt am Main, Germany 2014.

5. Hashemian, H. Wireless sensors for predictive maintenance of rotating equipment in research reactors. *Ann. Nucl. Energy* **2011**, *38*, 665–680. [CrossRef]
6. Low, K.S.; Win, W.; Er, M.J. Wireless Sensor Networks for Industrial Environments. In Proceedings of the International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06), Vienna, Austria, 28–30 November 2005; Volume 2, pp. 271–276.
7. Kumar S., A.A.; Ovsthus, K.; Kristensen, L.M. An Industrial Perspective on Wireless Sensor Networks—A Survey of Requirements, Protocols, and Challenges. *IEEE Commun. Surv. Tutorials* **2014**, *16*, 1391–1412. [CrossRef]
8. Das, K.; Zand, P.; Havinga, P. Industrial Wireless Monitoring with Energy-Harvesting Devices. *IEEE Internet Comput.* **2017**, *21*, 12–20. [CrossRef]
9. Hart Specification. Available online: https://fieldcommgroup.org/hart-specifications/ (accessed on 27 January 2025).
10. *IEC 62601:2011*; Industrial Communication Network—Fieldbus Specifications—Wia-pa Communication Network and Communication Profile. IEC: Geneva, Switzerland, 2011.
11. Qureshi, K.N.; Abdullah, A.H. Adaptation of wireless sensor network in industries and their architecture, standards and applications. *World Appl. Sci. J.* **2014**, *30*, 1218–1223.
12. Song, J.; Han, S.; Mok, A.; Chen, D.; Lucas, M.; Nixon, M.; Pratt, W. WirelessHART: Applying wireless technology in real-time industrial process control. In Proceedings of the 2008 IEEE Real-Time and Embedded Technology and Applications Symposium, St. Louis, MO, USA, 22–24 April 2008; pp. 377–386.
13. Zheng, L. Industrial wireless sensor networks and standardizations: The trend of wireless sensor networks for process autometion. In Proceedings of the SICE Annual Conference 2010, Taipei, Taiwan, 18–21 August 2010; pp. 1187–1190.
14. Montero, S.; Gozalvez, J.; Sepulcre, M.; Prieto, G. Impact of mobility on the management and performance of WirelessHART industrial communications. In Proceedings of the 2012 IEEE 17th International Conference on Emerging Technologies & Factory Automation (ETFA 2012), Krakow, Poland, 17–21 September 2012; pp. 1–4.
15. Kim, S.; Kim, C.; Cho, H.; Jung, K. A hierarchical routing graph for supporting mobile devices in industrial wireless sensor networks. *Sensors* **2021**, *21*, 458. [CrossRef] [PubMed]
16. Montero, S.; Gozalvez, J. LAN-ND, a new neighbour discovery protocol for mobile WirelessHART industrial networks. In Proceedings of the 2013 IEEE 18th Conference on Emerging Technologies & Factory Automation (ETFA), Cagliari, Italy, 10–13 September 2013; pp. 1–8.
17. Sun, W.; Yang, Z.; Zhang, X.; Liu, Y. Energy-efficient neighbor discovery in mobile ad hoc and wireless sensor networks: A survey. *IEEE Commun. Surv. Tutorials* **2014**, *16*, 1448–1459. [CrossRef]
18. Montero, S.; Gozalvez, J.; Sepulcre, M. Neighbor discovery for industrial wireless sensor networks with mobile nodes. *Comput. Commun.* **2017**, *111*, 41–55. [CrossRef]
19. Kim, S.; Kim, C.; Cho, H.; Jung, K.; Oh, S. Mobility Support Scheme Based on Machine Learning in Industrial Wireless Sensor Network. *Kips Trans. Comput. Commun. Syst.* **2020**, *9*, 256–264.
20. Alhulayil, M.; Al-Mistarihi, M.F.; Shurman, M.M. Performance Analysis of Dual-Hop AF Cognitive Relay Networks with Best Selection and Interference Constraints. *Electronics* **2023**, *12*, 124. [CrossRef]
21. Han, S.; Zhu, X.; Mok, A.K.; Chen, D.; Nixon, M. Reliable and real-time communication in industrial wireless mesh networks. In Proceedings of the 2011 17th IEEE Real-Time and Embedded Technology and Applications Symposium, Chicago, IL, USA, 11–14 April 2011; pp. 3–12.
22. Network Simulator-3.26. Available online: http://www.nsnam.org/ (accessed on 27 January 2025).
23. Zaitseva, E.; Rabcan, J.; Levashenko, V.; Kvassay, M. Importance analysis of decision making factors based on fuzzy decision trees. *Appl. Soft Comput.* **2023**, *134*, 109988. [CrossRef]