applied sciences

MDPI

Article

Efficient Node Insertion Algorithm for Connectivity-Based Multipolling MAC Protocol in Wi-Fi Sensor Networks

Woo-Yong Choi

Department of Industrial and Management Systems Engineering, Dong-A University, Busan 49315, Republic of Korea; wychoi77@dau.ac.kr

Abstract: Since low-power Wi-Fi sensors are connected to the Internet, effective radio spectrum use is crucial for developing an efficient Medium Access Control (MAC) protocol for Wi-Fi sensor networks. A connectivity-based multipolling mechanism was employed for Access Points to grant uplink transmission opportunities to Wi-Fi nodes with a reduced number of multipolling frame transmissions. The existing connectivity-based multipolling mechanism in IEEE 802.11 wireless LANs with many nodes may require excessive time to derive the optimal number of serially connected sequences due to the backtracking algorithm based on the Traveling Salesman Problem model. This limitation hinders the real-time implementation of the connectivity-based multipolling mechanism in Wi-Fi sensor networks. In this study, an efficient node insertion algorithm is proposed, by which the number of derived serially connected multipolling sequences that cover nodes in Wi-Fi sensor networks converges to only one as the number of Wi-Fi sensors increases in Wi-Fi sensor networks. As verified by simulation experiments for Wi-Fi sensor networks, the proposed node insertion algorithm produces a near-optimal number of multipolling sequences that cover the nodes in Wi-Fi sensor networks. This study proposes a node insertion algorithm for the real-time implementation of the connectivity-based multipolling mechanism in MAC protocol for Wi-Fi sensor networks.

Keywords: wireless connectivity; multipolling; MAC; sensor networks; wireless LANs

Citation: Choi, W.-Y. Efficient Node Insertion Algorithm for Connectivity-Based Multipolling MAC Protocol in Wi-Fi Sensor Networks. *Appl. Sci.* **2023**, *13*, 11974. https://doi.org/10.3390/app132111974

Academic Editors: Thomas Newe and Kashif Naseer Qureshi

Received: 28 August 2023
Revised: 27 October 2023
Accepted: 1 November 2023
Published: 2 November 2023

1. Introduction

Many wireless sensors must be connected to the Internet, ensuring that the static information of products to which sensors are attached and the dynamic sensing information of the environment where sensors are deployed are delivered and utilized for efficient decision making to realize a ubiquitous computing society. An increasing number of wireless sensors are connected to the Internet for public security and health data monitoring because of the development of wireless LAN technologies, such as IEEE 802.11ah and IEEE 802.11s, and other non-802.11 technologies. Developing an efficient Medium Access Control (MAC) protocol that harnesses the radio resource of wireless sensor networks with a large number of sensors is a vital part of the success of wireless sensor networks in the future.

The connectivity-based multipolling MAC protocol was developed to improve the efficiency of the multipolling MAC protocol by reducing the multipolling frame transmissions [1–4]. Using connectivity and interference information among nodes, a simultaneous polling method was proposed to allow multiple direct data communication between nodes to be carried out [1]. The backtracking algorithm was introduced to derive the connected multipolling sequence [2]. Frame aggregation and connectivity-based multipolling techniques were combined to efficiently collect Radio Frequency Identification (RFID) tag information from nodes [3]. The combined clustering and sequencing method was proposed for a heuristic approach for deriving the connected multipolling sequence of RFID readers [4]. However, in some cases, a scheduling algorithm based on the Traveling Salesman Problem (TSP) model, which is employed by the connectivity-based multipolling MAC protocol, requires too much time to obtain the minimal number of serially connected multipolling sequences in cases where a large number (e.g., >100) of nodes exist in wireless LANs [1–4]. An efficient real-time implementable algorithm should be developed for deriving serially connected multipolling sequences of nodes when a large number of nodes exist in wireless LANs.

In this study, an efficient node insertion algorithm is proposed by modifying the existing scheduling algorithm based on the TSP model to construct serially connected multipolling sequences of nodes for the connectivity-based multipolling MAC protocol. The next node can be selected to be added to a serially connected multipolling sequence based on the connectivities from recently added multiple nodes. The nodes simultaneously connected from the most recently added nodes are not skipped. They are selected to be added to the serially connected multipolling sequences using the proposed node insertion algorithm. Therefore, a smaller number of serially connected multipolling sequences of nodes can be derived to cover all Wi-Fi sensor nodes in a wireless LAN. In densely populated wireless LANs, a single serially connected multipolling sequence can be derived for all Wi-Fi sensors, which will be shown by the simulation results later in this paper. Although the previous scheduling algorithm may backtrack to obtain a smaller number of serially connected multipolling sequences, the proposed node-insertion algorithm does not backtrack to derive a near-optimal number of serially connected multipolling sequences in real time.

The existing backtracking algorithm has the computational complexity of $O(l!)$, where l is the number of sensors in a wireless sensor network, while obtaining the minimal number of serially connected multipolling sequences. The proposed node insertion algorithm almost eliminates the computational time uncertainty from the backtracking algorithm because the proposed node insertion algorithm has the computational complexity of $O(l^3)$ in deriving a near-optimal number of serially connected multipolling sequences. Second, the performance of the existing connectivity-based multipolling MAC protocol can be enhanced by the proposed node insertion algorithm, which derives a near-optimal number of serially connected multipolling sequences in real time. Finally, the proposed node insertion algorithm can accelerate the migration of Wi-Fi networks to wireless sensor networks. The contributions are summarized as follows:

- The node insertion algorithm almost eliminates the computational time uncertainty from the backtracking algorithm for deriving the minimal number of serially connected multipolling sequences.
- The performance of the connectivity-based multipolling MAC protocol is greatly enhanced by the node insertion algorithm.
- The node insertion algorithm can accelerate the migration of Wi-Fi networks to wireless sensor networks.

As can be seen in the next section, where the MAC protocols for wireless sensor networks in the literature [5–38] are surveyed, most of the MAC protocols employed in wireless sensor networks based on Wi-Fi networks are contention-based. However, to resolve the immense interference among many sensors, it is necessary to develop an efficient contention-free MAC protocol for wireless sensor networks based on Wi-Fi networks. The problem tackled in this paper is stated as follows:

- Developing a contention-free MAC protocol for wireless sensor networks based on Wi-Fi networks that allows Wi-Fi sensors to transmit their data with minimal transmission overhead.

2. Related Works

In IEEE 802.11 protocol [5], the technologies were extended to allow sensor grouping, minimizing their contention and enabling relay nodes to extend the communication range. IEEE 802.11 mesh networks were defined, and a hybrid routing protocol, which is a Hybrid Wireless Mesh Protocol, combining elements of the ad hoc On-demand Distance Vector and tree-based routing protocols, was proposed [6]. A new crowd-counting method has been proposed using wireless sensor and neural networks [7]. New analytical and simulation models were proposed for gathering medical information through wireless sensor networks [8].

Alnazir et al. [9] proposed the improved contention-based MAC protocols of Distributed Coordination Function and Enhanced Distributed Coordination Function for the application services of the Internet of Things (IoT). Choi [10] proposed a new clustering method for wireless sensor networks so that sensors can directly or indirectly communicate with each other in an energy-efficient manner. The service area of wireless sensor networks can be efficiently extended by appointing multiple appropriate pseudo-access points (APs) according to Choi [11]. The new design of full duplex MAC protocol for 5G wireless LANs was proposed by Gupta and Venkatesh [12]. In a study conducted by CV and Sathish [13], an underwater wireless sensor network was demonstrated using wireless LAN technology. From the survey of the research on wireless sensor networks based on Wi-Fi networks, it is observed that most of the MAC protocols employed in wireless sensor networks are contention-based and govern communication among less than 100 sensors.

Now, the research on wireless sensor networks based on non-802.11 infrastructure networks will be reviewed. Jubair et al. [14] compared and reviewed various techniques and protocols for optimizing cluster-based sensor networks. The new energy-efficient routing protocols for locating objects or gathering sensor information using wireless sensor networks are proposed in two recent studies [15,16]. The hybrid algorithm, Hybrid Pigeon Inspired with Glowworm Swarm Optimization, is proposed to optimize clustering wireless sensor networks [17]. Romli et al. [18] used the unmanned aerial vehicle to gather data from wireless sensors distributed on the ground. The multichannel MAC protocol that disperses the wakeup times of neighboring wireless sensors was proposed to reduce data collisions [19]. The synchronized power-saving mechanism and modified backoff algorithm were presented to reduce energy consumption and data collisions in Ultra Wide Band wireless sensor networks [20]. The S-MAC protocol was enhanced to improve the quality of service (QoS) of wireless sensor networks [21]. The contention-based MAC protocol was introduced to reduce power consumption and data collisions and mitigate the hidden terminal problem in wireless sensor networks [22]. The MAC protocol proposed by Yang et al. [23] reduces energy consumption by the sender's prediction of the receiver's wakeup time. Receiver-initiated power saving and multipriority backoff mechanisms were employed to reduce energy consumption and improve the QoS of wireless sensor networks [24]. The cross-layer MAC protocol utilizes the metadata of sensors to reduce energy consumption in wireless sensor networks [25]. Alanazi proposed the new synchronized power-saving mechanism for wireless sensor networks [26]. The cross-layer design approach of the MAC protocol was proposed to optimize energy consumption by the synchronized Carrier Sense Multiple Access (CSMA) mechanism [27]. The receiver-initiated and cooperative MAC protocol was presented to reduce energy consumption and data collisions in wireless sensor networks [28].

In addition to the contention-based MAC protocols, in the literature, reservation-based or contention-free MAC protocols can be found for wireless sensor networks. The contention-free and low-delay MAC protocol was proposed for wireless sensor networks in underwater environments [29]. The distributed Time Division Multiple Access (TDMA)-based MAC protocol was employed to mitigate data collisions in vehicular ad hoc networks [30]. The channel reservation mechanism through Request-to-Send (RTS) and Test-to-Transmit handshake was proposed for a secured communication in an intelligent 6G network [31]. The link delay aware MAC protocol based on TDMA and Frequency Division Multiple Access was designed to improve energy consumption in wireless sensor networks [32]. TDMA schedule-based MAC protocols were used for energy harvesting in cognitive wireless sensor networks [33,34]. The hybrid CSMA/TDMA MAC protocol was presented to satisfy a variety of QoS requirements in wireless sensor networks [35]. The TDMA-based hybrid MAC protocol was proposed to optimally control a power-saving mechanism to reduce energy consumption in wireless sensor networks [36]. The reservation-based MAC protocol through the scheduling of RTS frames was employed to reduce data collisions and energy consumption for wireless sensor networks [37]. The reservation-based MAC protocol through the random transmissions of RTS frames used Nash equilibrium theory to improve the throughput of wireless sensor networks [38]. In the case of wireless sensor networks based on non-802.11 networks like Bluetooth networks, the research efforts are divided into contention-based, contention-free or reservation-based approaches, and recently, more research efforts have focused on the development of contention-free or reservation-based MAC protocols for wireless sensor networks.

We can summarize the research trend on wireless sensor networks based on Wi-Fi networks and non-802.11 networks, as shown in Table 1.

Table 1. Trend of research on wireless sensor networks.

		MAC Method	
		Contention-Based	**Contention-Free or Reservation-Based**
Infrastructure Network	Wi-Fi Network	Most	Little
	Non-802.11 Network	Fair	Fair

The MAC protocol's performance significantly degrades as the number of sensors in wireless sensor networks increases due to the interference among sensors. When it comes to wireless sensor networks based on non-802.11 networks like Bluetooth networks, where sensors have a short transmission range, this degradation of network performance can be multiplied as data go through multi-hop links. This may lead to the early adoption of contention-free or reservation-based MAC protocols for wireless sensor networks based on non-802.11 networks. Research on contention-free or reservation-based MAC protocols for wireless sensor networks based on Wi-Fi networks is crucial for enhancing global network performance in a 6G society. Therefore, it is necessary to develop a contention-free MAC protocol that allows many Wi-Fi sensors to transmit their uplink data with minimal transmission overhead to utilize the radio spectrum more efficiently in Wi-Fi sensor networks.

3. Connectivity-Based Multipolling MAC Protocol

This section will briefly explain the mechanism of the connectivity-based multipolling MAC protocol [1–4]. The connectivity-based multipolling MAC protocol is based on the Point Coordination Function (PCF) [39]. According to the PCF, APs transmit a separate polling frame to each node to grant uplink transmission opportunity. However, when a large number of nodes exist in wireless LANs, and the uplink MAC frames have a small payload, which is the case for wireless LANs with a large number of Wi-Fi sensors having relatively a small amount of static and dynamic sensing information, the MAC overhead of the separate polling transmissions for all nodes becomes significant. The connectivity-based multipolling MAC protocol was proposed to resolve this problem. The connectivity-based multipolling MAC protocol mainly focuses on optimizing the general uplink data transmission procedure in Wi-Fi sensor networks with multipolling frame transmissions but not on the query-based information search for a specific sensor or a group of sensors.

Initially, when APs have not yet collected connectivity information among nodes, the connectivity-based multipolling MAC protocol is operated according to the PCF. The connectivity information among nodes indicates whether each node can successfully hear and decode the transmission signals from other nodes. Each node i responds to APs by piggybacking on its uplink data or null frame whether it can hear the transmission signals from each other node j, that is, node i is connected from node j, overhearing the transmission signals from node j. From the collected connectivity information among nodes, APs construct serially connected multipolling sequences of nodes that cover all nodes that are associated with themselves.

When APs multicast a multipolling frame to the nodes in a serially connected multipolling sequence, in which the recipient MAC addresses are specified in the MAC header of the multipolling frame, each recipient node knows its order in the sequence. The first recipient transmits its data or null frame in a short interframe space (SIFS) period after the reception of the multipolling frame, and other nodes in the sequence transmit their data or null frames in an SIFS period after receiving the transmission signals from the node of the previous order in the sequence. Note that each node, besides the first recipient, can hear the transmission signals from the previous recipient in a serially connected multipolling sequence. Each node responds to APs with updated connectivity information during the connectivity-based multipolling process, which is the change in the set of other nodes from which it is connected. APs should update the serially connected multipolling sequences based on the updated connectivity information among nodes. Based on the mechanism of each sensor reporting to APs, the change in connectivity information, and APs updating the serially connected multipolling sequences based on the updated connectivity information among nodes, the dynamic nature of Wi-Fi sensor networks can be reflected in the connectivity-based multipolling MAC protocol.

When a node does not respond to a multipolling frame, the APs should retransmit the multipolling frame for error recovery, where the multipolling sequence should include only the partial sequence of the original sequence after the failed node. When the failed node is not the first node in the multipolling sequence, the APs consider the connectivity from the previous node of the failed node to the failed node to be broken. For efficient and reliable uplink data transfer by the connectivity-based multipolling MAC protocol, the bitmap technique can assign a bit to each recipient MAC address in the multipolling frames. The APs can indicate whether the previous uplink data transfer from the node of the corresponding MAC address was successful by setting the bit assigned to a recipient MAC address to zero or one.

The connectivity-based multipolling process consists of two phases: the initialization phase and the multipolling phase. During the initialization phase using the PCF, APs poll each sensor in a round-robin manner, and each sensor responds with the data or null frame on which the connectivity information is piggybacked. Using the connectivity information collected from sensors, the APs initiate the multipolling phase by transmitting the multipolling frames, and the sensors polled by the multipolling frames respond with their data or null frames, on which the change in the connectivity information is piggybacked, in a serially connected manner. This process is depicted in Figure 1. In Figure 1b, node 3 hears the transmission signals from node 2, while node 1 hears the transmission signals from node 3, and so on.

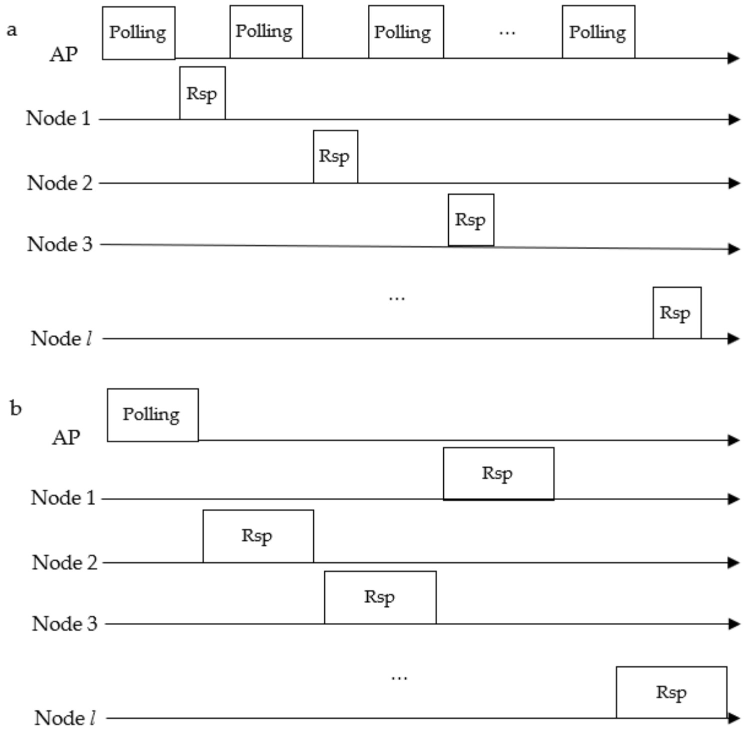

Figure 1. Connectivity-based multipolling process. (**a**) Initialization phase. (**b**) Multipolling phase. Rsp represents response frame.

4. Efficient Node Insertion Algorithm

4.1. Motivational Idea

When sequencing nodes in a serially connected manner, the previous scheduling algorithm employed by the connectivity-based multipolling MAC protocol considered only the connectivity from the most recently added node in a sequence to select the next node. This may result in selecting the next node not connected to other recently added nodes in the sequence by skipping nodes simultaneously connected to multiple recently added nodes. Therefore, as the scheduling algorithm progresses, the remaining nodes that have not been added to the sequences may not have sufficient connectivity; thus, the scheduling algorithm may fail to construct a minimal number of serially connected multipolling sequences for the remaining nodes in real time. For example, Figure 2 shows that ten nodes 1, 2, . . ., and 10 are deployed in a row, and nodes i and j for $|i - j| < 3$ are mutually connected. Let us consider only the connectivity from the most recently added node in the sequence to insert a new node into a serially connected multipolling sequence. Three serially connected multipolling sequences: (1, 3, 4, 6, 7, 9, 10, and 8), (2), and (5) can be constructed. The scheduling algorithm randomly selects the next nodes among the nodes that are connected from the most recently added nodes in the first multipolling sequence. For example, after the insertion of node 1, the scheduling algorithm randomly selects node 3 among nodes 2 and 3, which are connected from node 1. After constructing the first multipolling sequence (1, 3, 4, 6, 7, 9, 10, and 8), the remaining nodes 2 and 5 are not connected. However, a single serially connected multipolling sequence (1, 2, 3, 4, 5, 6, 7, 8, 9, and 10) can cover ten nodes. Although the scheduling algorithm can find the minimal number of serially connected multipolling sequences by a backtracking search in the end, as the number of nodes in a wireless LAN increases, the possibility of the previous scheduling algorithm being able to find the minimal number of serially connected multipolling sequences in real time significantly decreases.

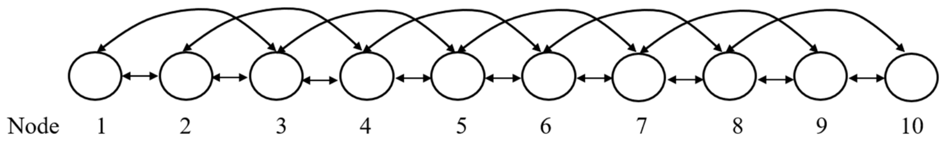

Figure 2. Example of node deployment in a row. Each circle represents a node, and each bidirectional arrow represents mutual connectivity between a pair of nodes.

The optimization problem for constructing the minimal number of serially connected multipolling sequences that covers a set of nodes can generally be formulated as a TSP [1–4]. When a relatively small number of nodes exist in wireless LANs, it is beneficial to solve the general TSP problem using a backtracking algorithm. However, when many nodes exist in wireless LANs, which may be the case with Wi-Fi sensor networks, solving the TSP problem using a backtracking algorithm can be extremely time consuming in some cases. As the number of Wi-Fi sensor nodes grows in wireless LANs, the connectivity from the most recently added multiple nodes in a serially connected multipolling sequence in construction should be considered to insert a new node into the sequence so that the search space becomes smaller. The nodes simultaneously connected from the most recently added multiple nodes are not skipped from being inserted into the serially connected multipolling sequence in construction. Moreover, to save the time required by the newly proposed node insertion algorithm, it does not backtrack or undo any insertion of the previous node in sequence. The proposed node insertion algorithm is explained in the following subsection.

4.2. Description of Algorithm

We assume a wireless LAN consisting of an AP and Wi-Fi sensors associated with the AP. Each low-power Wi-Fi sensor was assumed to be able to transmit its data frame directly to the AP at a relatively low transmission speed, as in [7]. The connectivity-based multipolling MAC protocol was employed for each Wi-Fi node to transmit its uplink data frames to the AP. In this subsection, the node insertion algorithm is explained, by which the serially connected multipolling sequences of Wi-Fi sensor nodes necessary for the connectivity-based multipolling MAC protocol are derived.

An unselected node can be randomly selected as the first Wi-Fi node of a new serially connected multipolling sequence. When inserting a new Wi-Fi node into the serially connected multipolling sequence of nodes being constructed, the primary condition for selecting the next Wi-Fi node is that it should be connected from the most recently added node in the sequence. However, suppose multiple Wi-Fi nodes connected from the most recently added node are available. In that case, a secondary condition is proposed for selecting the next Wi-Fi node. After selecting the new Wi-Fi node for the sequence, the connectivity of the remaining Wi-Fi nodes that have not been selected is maximized. The connectivity of a set S of nodes is defined as the number of ordered pairs (i, j) such that node j is connected from node i, $i \neq j$, and i and j are in S. If no new Wi-Fi node satisfies the first condition, the current serially connected multipolling sequence is closed, and a new serially connected multipolling sequence of nodes begins to be constructed by selecting the random remaining Wi-Fi node as the first node of the new serially connected multipolling sequence. Let us consider only the first condition for selecting the next Wi-Fi node to be added to the serially connected multipolling sequence. A minimal number of serially connected multipolling sequences may not be constructed for the remaining Wi-Fi nodes because the connectivity of the set of the remaining Wi-Fi nodes may not be maximized. Note that in Figure 2, if node 4 is selected as the next node after the addition of nodes 1 and 3 to the first serially connected multipolling sequence, the connectivity of the set of remaining nodes 2, 5, 6, . . . , 10 is not maximized because node 2 is not connected to any other remaining node.

The easy way of implementing the primary and secondary conditions for selecting the next Wi-Fi node that is to be inserted into a serially connected multipolling sequence in construction is that the Wi-Fi nodes are searched that are simultaneously connected from the m (=1, 2, . . .) most recently added nodes in the sequence. As the next node, a Wi-Fi node is selected that is simultaneously connected from as many most recently added nodes in the sequence as possible. By selecting the next node, the selected Wi-Fi node is most likely to be connected from the smallest number of nodes that have not been selected so that the connectivity of the set of the remaining nodes that have not been selected is maximized. Note that in Figure 2, after the addition of nodes 1 and 3 to the first serially connected multipolling sequence in the construction, node 2 has the largest number of nodes in the current sequence from which it is connected and has the smallest number of unselected nodes from which it is connected, so that the connectivity of the set of the remaining nodes 4, 5, 6, . . ., and 10 is maximized.

m should not exceed the number of Wi-Fi nodes in the current sequence during construction. If, with $m = 1$, no Wi-Fi node is available that is connected from $m = 1$, the most recently added node in the sequence, the current serially connected multipolling sequence that is being constructed should be closed, and a new serially connected multipolling sequence should begin to be constructed for the remaining Wi-Fi nodes that have not been selected. If m = a certain value n (>0), the Wi-Fi nodes are found that are simultaneously connected from the $m = n$ most recently added nodes in the sequence, and the Wi-Fi nodes are searched that are simultaneously connected from the $m = n + 1$ most recently added nodes in the sequence. If with $m = n$ (>1), no Wi-Fi node is found that is simultaneously connected from the $m = n$ most recently added nodes in the sequence, to the current serially connected multipolling sequence, a random Wi-Fi node should be inserted that is simultaneously connected from the $m = n - 1$ (>0) most recently added nodes in the sequence. Continuing to construct serially connected multipolling sequences by inserting new Wi-Fi nodes that are simultaneously connected from as many of the most recently added nodes in the current sequences as possible to the serially connected multipolling sequences that are being constructed, the final serially connected multipolling sequences can be derived that cover the Wi-Fi sensor nodes in the wireless LAN.

Applying the aforementioned node insertion algorithm to ten nodes in Figure 2, after inserting nodes 1 and 3 into the first serially connected multipolling sequence, node 2 should be inserted as the next node to the first sequence. Then, nodes 4, 5, 6, 7, 8, 9, and 10 are sequentially inserted, and a single serially connected multipolling sequence (1, 3, 2, 4, 5, 6, 7, 8, 9, 10) that covers ten nodes can be constructed.

The pseudocode and the flow chart for implementing the proposed node insertion algorithm are presented in Algorithm 1 and Figure 3, respectively.

In Algorithm 1 and Figure 3, the top loop is for selecting the next node, which is a node that is simultaneously connected from as many of the most recently added nodes in the current sequence as possible until the derived multipolling sequences cover all nodes. The second top loop sets the values of *possibility*[i][j] at each stage of selecting the next node. *possibility*[i][j] is set to 1 if node j is connected from the previous nodes up to the last (number of sensors in the current multipolling sequence in construction $- i + 1)^{th}$ node, and *depth* is set to the minimal value of i such that there are one or more j values satisfying the condition of *possibility*[i][j] = 1. Note that any node j such that *possibility*[*depth*][j] = 1 is a node that is simultaneously connected to as many of the most recently added nodes in the current sequence as possible.

When applying the proposed node insertion algorithm to the network in Figure 2 to derive the serially connected multipolling sequence (1, 3, 2, 4, 5, 6, 7, 8, 9, 10), the process of setting the *possibility*[i][j] values and selecting the next nodes to be inserted into the multipolling sequence during construction is depicted in Figure 4.

Algorithm 1: Pseudocode of node insertion algorithm

for(n = 1; $n \leq$ # of sensors; n++)
  for(i = # of sensors in current multipolling seq.; $i \geq 1$; i--)
    if(i = # of sensors in current seq.)
      possibility[i][j] = 1 for all unselected j's such that node j is connected from the previous node;
      possibility[i][j] = 0 for all other j's;
    else
      possibility[i][j] = 1 for all unselected j's such that node j is connected from (# of sensors in current seq. − i + 1)th last node and possibility[i + 1][j] = 1;
      possibility[i][j] = 0 for all other j's;
    if(there is no j such that possibility[i][j] = 1)
      break and set the *depth* to the current stopped value of i + 1;
  if(there is no break in second top loop)
      set *depth* to 1;
  if(current seq. is empty)
    select any unselected node j and insert such node j to the current multipolling seq.;
  else if(*depth* > # of sensors in current multipolling seq.)
    close current multipolling seq. and start new multipolling seq. with any unselected node j;
  else
    select any j such that *possibility*[*depth*][j] = 1 and insert such node j to current multipolling seq.;

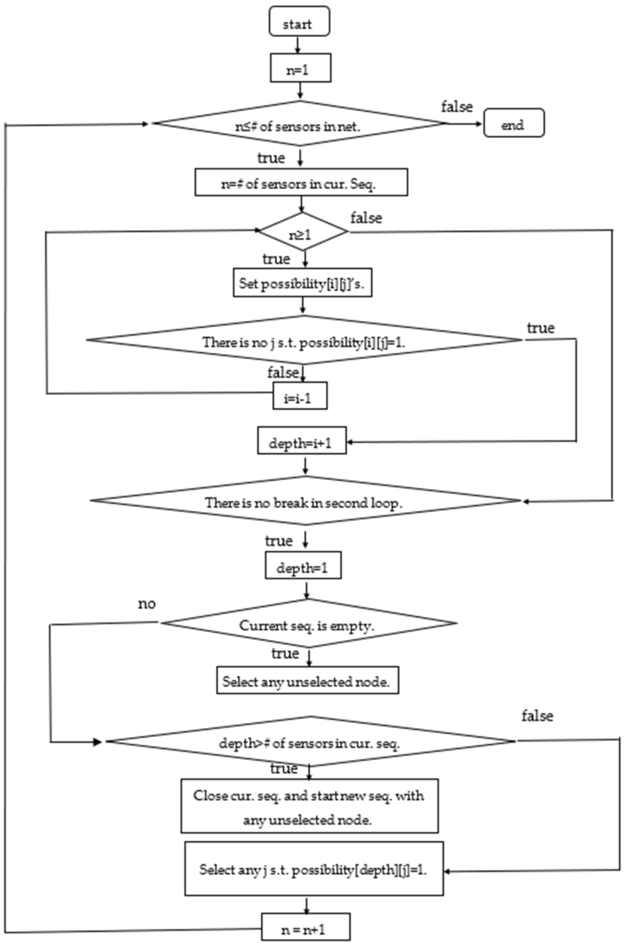

Figure 3. Flow chart of node insertion algorithm.

n = 1: node 1 selected.
n = 2: possibility[1][1] = 0, possibility[1][2] = possibility[1][3] = 1, possibility[1][4] = possibility[1][5] = ... = possibility[1][10] = 0.
→ depth 1, node 3 is selected.
n = 3: possibility[2][1] = 0, possibility[2][2] = 1, possibility[2][3] = 0, possibility[2][4] = possibility[2][5] = 1, possibility[2][6] = possibility[2][7] = ... = possibility[2][10] = 0,
possibility[1][1] = 0, possibility[1][2] = 1, possibility[1][3] = possibility[1][4] = ... possibility[1][10] = 0.
→ depth = 1, node 2 is selected.
n = 4: possibility[3][1] = possibility[3][2] = possibility[3][3] = 0, possibility[3][4] = 1, possibility[3][5] = possibility[3][6] = ... = possibility[3][10] = 0,
possibility[2][1] = possibility[2][2] = possibility[2][3] = 0, possibility[2][4] = 1, possibility[2][5] = possibility[2][6] = ... = possibility[2][10] = 0,
possibility[1][1] = possibility[1][2] = ... = possibility[1][10] = 0.
→ depth = 2, node 4 is selected.
n = 5: possibility[4][1] = possibility[4][2] = ... = possibility[4][4] = 0, possibility[4][5] = possibility[4][6] = 1, possibility[4][7] = possibility[4][8] = ... = possibility[4][10] = 0,
possibility[3][1] = possibility[3][2] = ... = possibility[3][10] = 0.
→ depth = 4, node 5 is selected.
n = 6: possibility[5][1] = possibility[5][2] = ... = possibility[5][5] = 0, possibility[5][6] = possibility[5][7] = 1, possibility[5][8] = possibility[5][9] = possibility[5][10] = 0,
possibility[4][1] = possibility[4][2] = ... = possibility[4][5] = 0, possibility[4][6] = 1, possibility[4][7] = possibility[4][8] = ... = possibility[4][10] = 0,
possibility[3][1] = possibility[3][2] = ... = possibility[3][10] = 0.
→ depth = 4, node 6 is selected.
n = 7: possibility[6][1] = possibility[6][2] = ... = possibility[6][6] = 0, possibility[6][7] = possibility[6][8] = 1, possibility[6][9] = possibility[6][10] = 0,
possibility[5][1] = possibility[5][2] = ... = possibility[5][6] = 0, possibility[5][7] = 1, possibility[5][8] = possibility[5][9] = possibility[5][10] = 0,
possibility[4][1] = possibility[4][2] = ... = possibility[4][10] = 0.
→ depth = 5, node 7 is selected.
n = 8: possibility[7][1] = possibility[7][2] = ... = possibility[7][7] = 0, possibility[7][8] = possibility[7][9] = 1, possibility[7][10] = 0,
possibility[6][1] = possibility[6][2] = ... = possibility[6][7] = 0, possibility[6][8] = 1, possibility[6][9] = possibility[6][10] = 0,
possibility[5][1] = possibility[5][2] = ... = possibility[5][10] = 0.
→ depth = 6, node 8 is selected.
n = 9: possibility[8][1] = possibility[8][2] = ... = possibility[8][8] = 0, possibility[8][9] = possibility[8][10] = 1,
possibility[7][1] = possibility[7][2] = ... = possibility[7][8] = 0, possibility[7][9] = 1, possibility[7][10] = 0,
possibility[6][1] = possibility[6][2] = ... = possibility[6][10] = 0.
→ depth = 7, node 9 is selected.
n = 10: possibility[9][1] = possibility[9][2] = ... = possibility[9][9] = 0, possibility[9][10] = 1,
possibility[8][1] = possibility[8][2] = ... = possibility[8][9] = 0, possibility[8][10] = 1,
possibility[7][1] = possibility[7][2] = ... = possibility[7][10] = 0.
→ depth = 8, node 10 is selected.

Figure 4. Process of selecting next nodes in node insertion algorithm.

5. Simulation Results

We assume that the service area of a Wi-Fi sensor network is circular, that an AP is located at the center of the service area, and l = 10, 20, 30, ... 100, 200, 400, 600, 800, or 1000 Wi-Fi sensors are associated with the AP and randomly located in the circular service area. Many smart factories have over 1000 sensors for the realization of IoT technology [40,41]. To cover this number of sensors by sensor networks, each sensor network should interconnect with up to 1000 sensors. The radius of the service area is denoted by r, and the transmission range of each Wi-Fi sensor, for which the radius $R = r$, $1.1r$, $1.2r$, $1.3r$, $1.4r$, or $1.5r$, is assumed to be circular. The simulation parameters for the number of sensors in the sensor network, and the transmission range of each sensor, their values, and other details of simulation setup are presented in Table 2. Three sample Wi-Fi sensor networks with l = 10, 20, and 30 Wi-Fi sensors and $R = r$ are generated to closely observe the behavior of the proposed node insertion algorithm, as shown in Figures 5–7. Applying the proposed node insertion algorithm to the three sample networks in Figures 5–7, serially connected multipolling sequences are constructed for the Wi-Fi sensors, and each constructed serially connected multipolling sequence is presented using a connected arrow.

Table 2. Simulation parameters, values and details of simulation setup.

Parameters	Values
Number of Sensors (l)	10, 20, 30, . . ., 100, 200, 400, 600, 800, 1000
Transmission Range (R)	r, 1.1r, 12r, 1.3r, 1.4r, 1.5r
Shape of Service Area	Circular
Location of AP	Center of Service Area
Location of Sensors	Uniformly Randomized

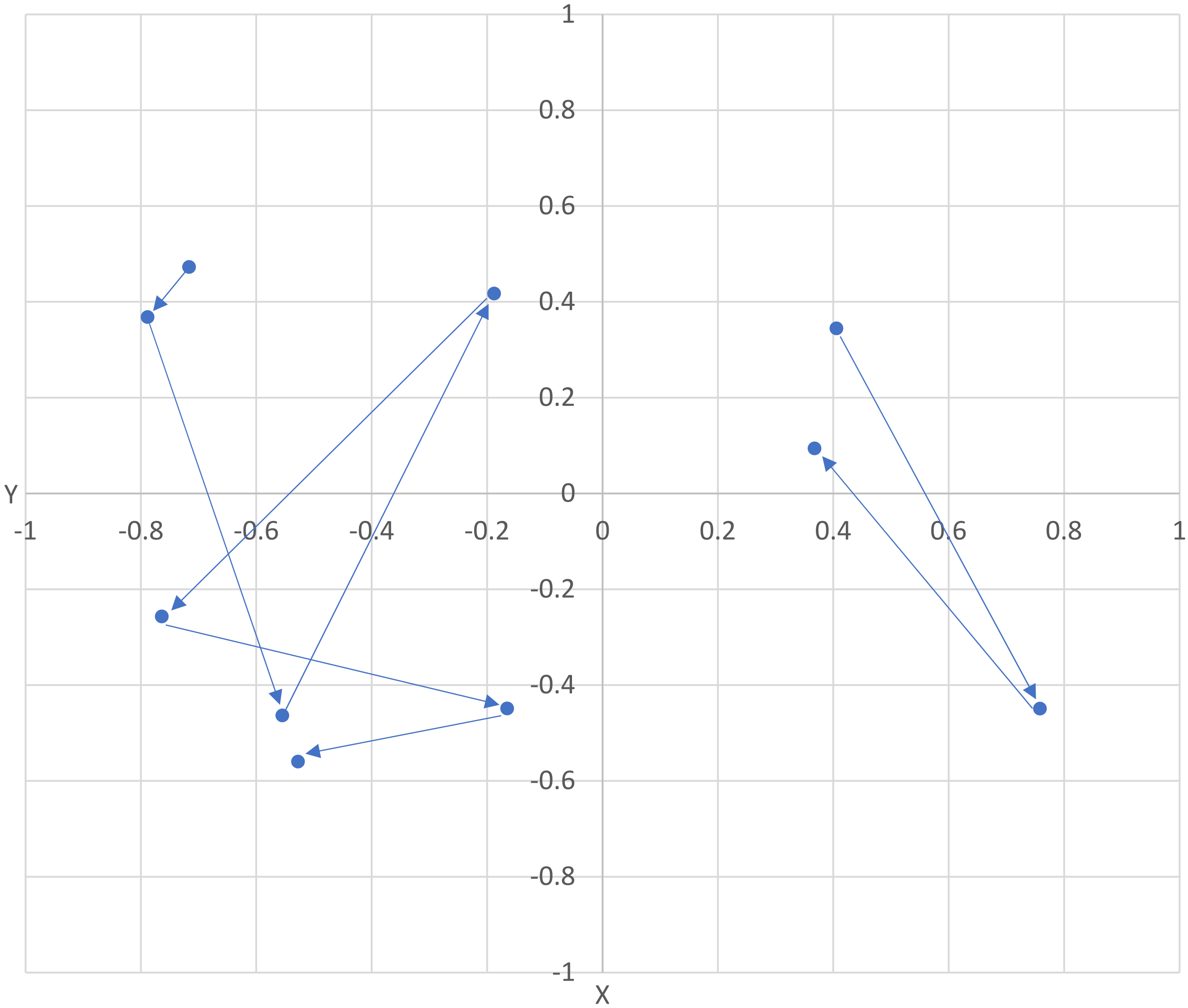

Figure 5. Sample Wi-Fi sensor network and formations of serially connected multipolling sequences with l = 10 and R = r.

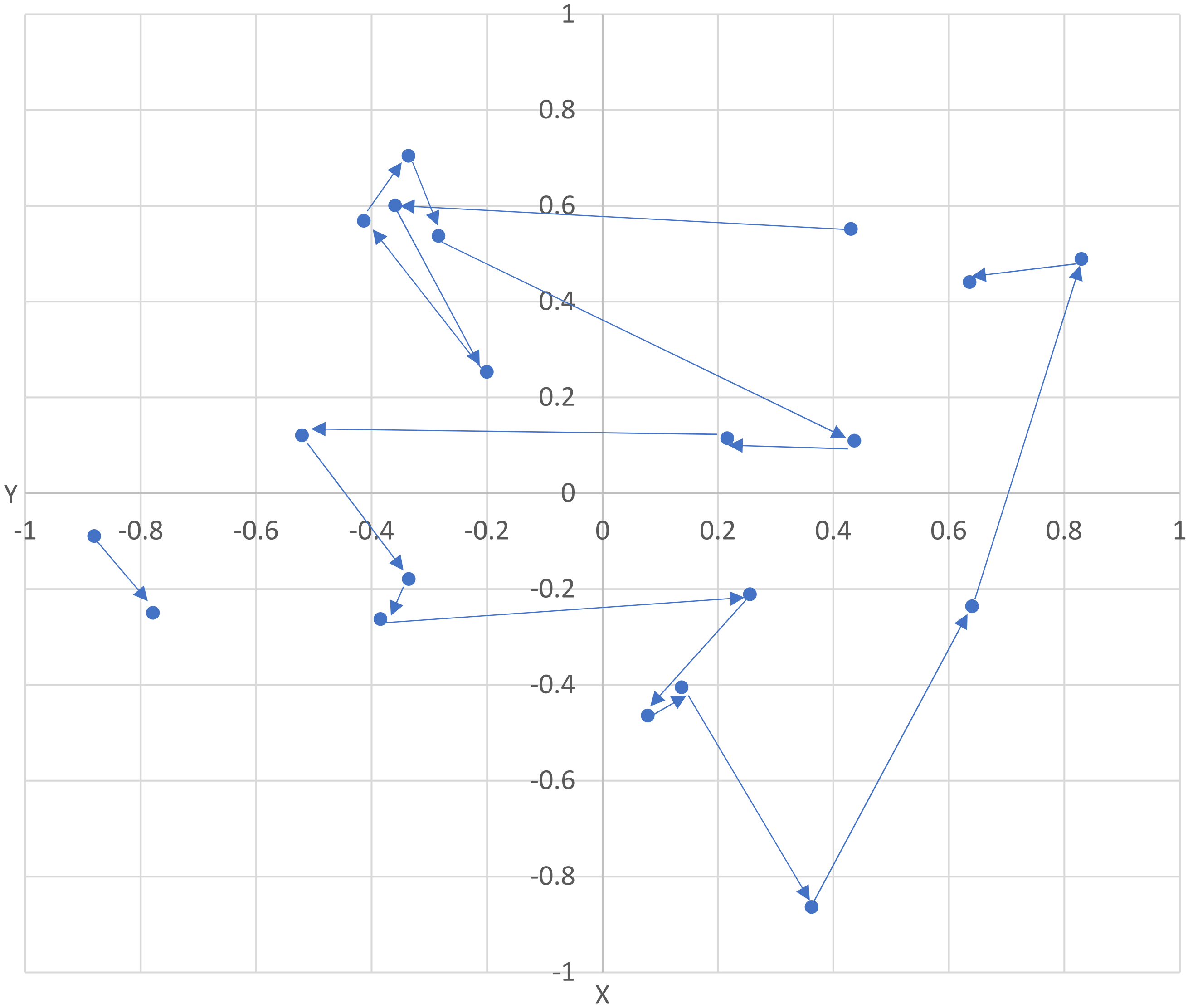

Figure 6. Sample Wi-Fi sensor network and formations of serially connected multipolling sequences with l = 20 and R = r.

In Figures 5–7, the coordinate value (0, 0) is the center of the circular service area, and the other coordinate values are relative to the radius r of the service area. Since the coordinate value of the location of each sensor and the transmission range (R) are relative to the radius r of the service area, that is, the coordinate value of the location of each sensor and the transmission range are simultaneously proportional to the radius r of the service area, the value of $r > 0$ does not affect the operation of the proposed node insertion algorithm or the simulation results. As shown in Figures 5–7, two serially connected multipolling sequences for each sample network with 10 and 20 Wi-Fi sensor nodes were derived, and only a single serially connected multipolling sequence for the sample network with 30 Wi-Fi sensor nodes was derived using the proposed node insertion algorithm. It can be seen that the algorithm tries its best not to skip the nodes simultaneously connected from the most recently added nodes in the sequences from the behavior of the proposed node insertion algorithm for sequencing nodes to construct serially connected multipolling sequences.

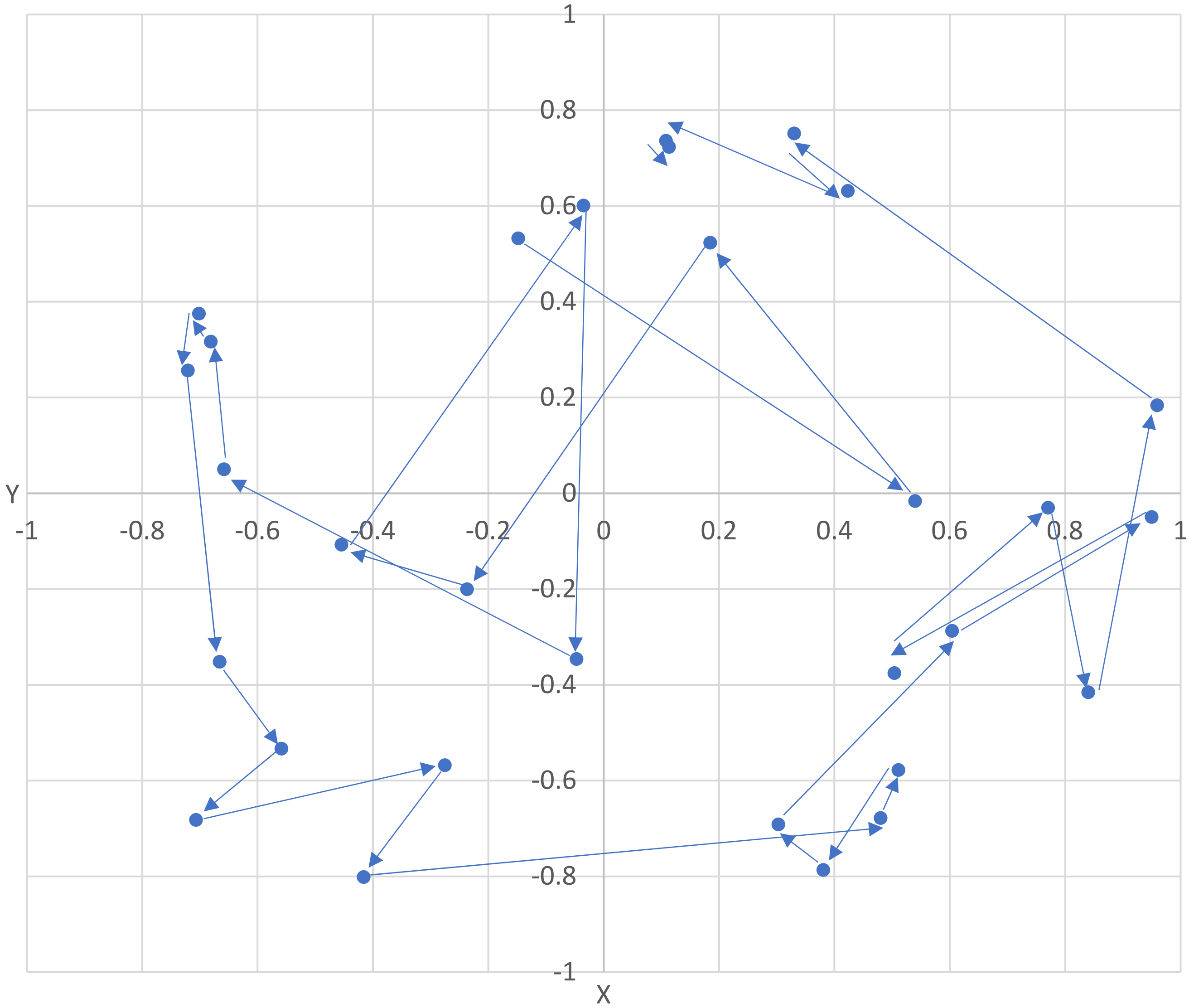

Figure 7. Sample Wi-Fi sensor network and formation of serially connected multipolling sequence with $l = 30$ and $R = r$.

We generated 50 random networks to obtain the performance of the proposed node insertion algorithm for the mean number of serially connected multipolling sequences for Wi-Fi sensor networks for each combined case of $l = 10, 20, 30, \ldots, 100, 200, 400, 600, 800$, or 1000 Wi-Fi sensors, and $R = r$, $1.1r$, $1.2r$, $1.3r$, $1.4r$ or $1.5r$. The mean number of serially connected multipolling sequences for each combined case is presented in Figure 8.

As shown in Figure 8, the proposed node insertion algorithm performs excellently in deriving, on average, fewer than two serially connected multipolling sequences that cover the Wi-Fi nodes in all cases. It can be seen that the number of serially connected multipolling sequences derived by the proposed node insertion algorithm tends to decrease as the number of Wi-Fi sensors increases in the sensor networks and converges to only one as the number of nodes increases in the sensor networks. This significant reduction in the multipolling frame transmission overhead results in each sensor being granted more wireless bandwidth to transmit uplink data, extending the lifetime of Wi-Fi sensor networks.

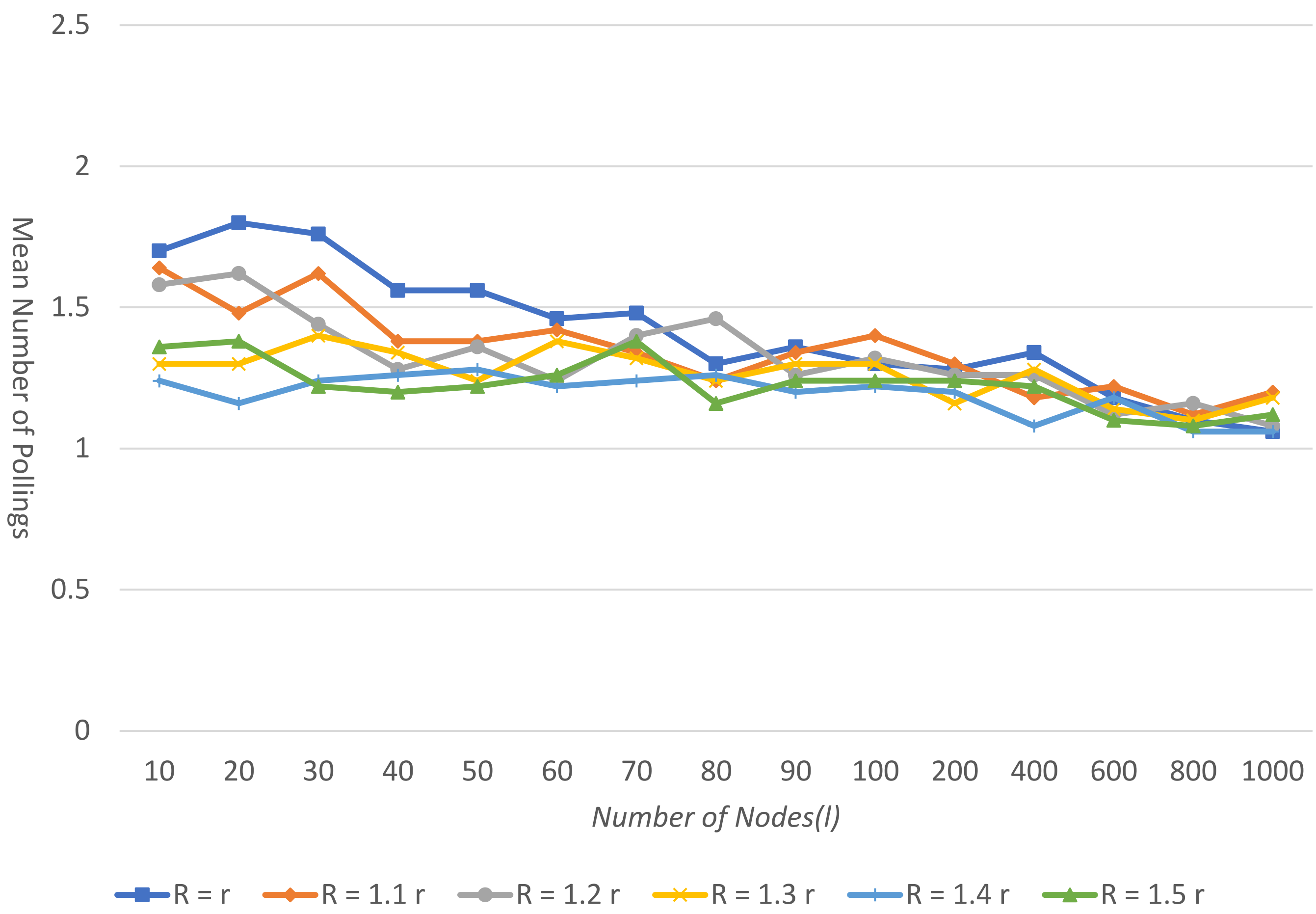

Figure 8. Results of mean numbers of serially connected multipolling sequences.

In Figure 9, the mean number of previously added nodes in the sequences is presented, from which the connectivities are considered for the proposed node-insertion algorithm to select the next nodes. As the number of previously added nodes from which the connectivities are considered to select the next nodes increases, the proposed node-insertion algorithm requires more time to derive serially connected multipolling sequences that cover the nodes. The proposed node insertion algorithm requires more time to derive serially connected multipolling sequences as the number l of nodes and the transmission range R increase. For the worst case of $l = 1000$ and $R = 1.5r$, the proposed node insertion algorithm took less than 0.8 s to derive the serially connected multipolling sequences that cover the nodes for each case of 50 random Wi-Fi sensor networks using a computer of 1.6 GHz CPU and 8 GB RAM.

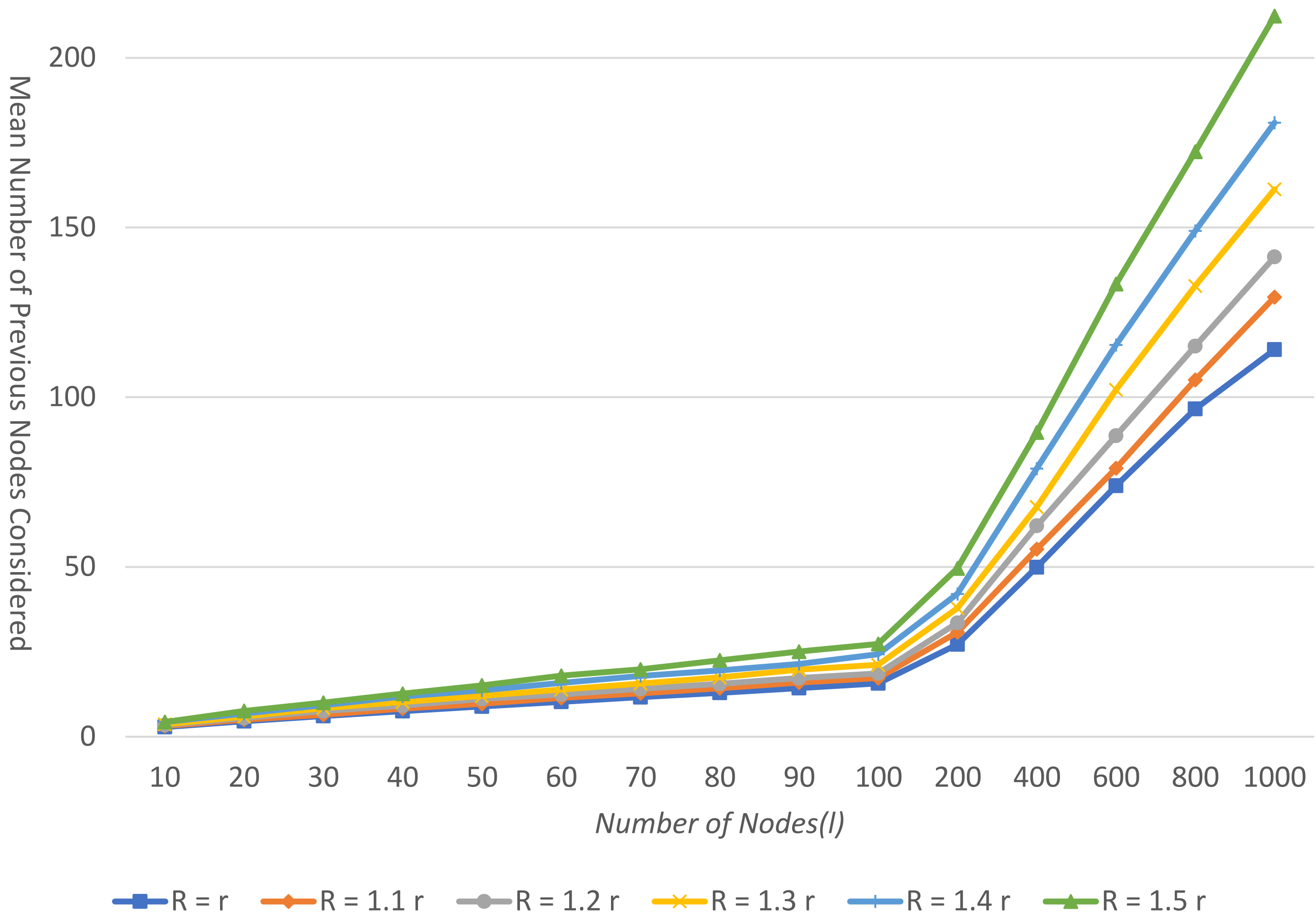

Figure 9. Results of considered mean numbers of previous nodes.

6. Discussion

Generally, when an existing connectivity-based multipolling mechanism based on the TSP model is applied to derive the optimal number of serially connected multipolling sequences for wireless LANs, the number of searches required by the existing connectivity-based multipolling mechanism increases as the number of nodes increases in wireless LANs [1–4]. Specifically, through simulation experiments conducted over more than 1000 random networks with l = 1000 sensors and transmission range $R = r$, 1.1r, 1.2r, 1.3r, 1.4r or 1.5r, the search algorithm based on the TSP model was found to take on average more than 14.67 s to derive the optimal number of multipolling sequences for each network and to take more than ten minutes to derive the optimal number of multipolling sequences in some cases of networks. A new node insertion algorithm is proposed to resolve the issue of excessive time taken by the existing connectivity-based multipolling mechanism in which the search space for the optimal number of serially connected multipolling sequences is dramatically reduced such that the number of real-time derived serially connected multipolling sequences converges to only one as the number of sensors in Wi-Fi sensor networks increases. When the number of sensors in Wi-Fi sensor networks is equal to 1000, the mean numbers of serially connected multipolling sequences are equal to 1.06, 1.2, 1.08, 1.18, 1.06, and 1.12 for $R = r$, 1.1r, 1.2r, 1.3r, 1.4r and 1.5r, respectively, according to the simulation experiments in Section 5. The strong result—that in real time, almost only one serially connected multipolling sequence can be derived for a Wi-Fi sensor network with many sensors—makes the MAC protocol based on the connectivity-based multipolling mechanism more attractive for Wi-Fi sensor networks.

From Figure 4, it can be seen that as new nodes are selected, *depth* does not decrease, which provides a clue for further reducing the search space. The proposed node insertion algorithm was found to be more efficiently implemented using one or more additional techniques for reducing the search space, such that serially connected multipolling sequences could be derived within approximately 15 ms for each simulated case in Section 5. A subsequent paper will be prepared and submitted for publication for a detailed implementation of techniques to reduce the search space further.

Now, why only one serially connected multipolling sequence can exist for the connectivity-based multipolling mechanism in a Wi-Fi sensor network with many sensors is discussed. The connectivity-based multipolling mechanism is based on the connectivity among nodes in wireless LANs. The connectivity-based multipolling mechanism aims to find the minimal number of serially connected multipolling sequences that cover all the nodes in wireless LANs. The performance of the search algorithm for the minimal number of serially connected multipolling sequences, which is the number of derived serially connected multipolling sequences, depends heavily on the connectivity of the set of nodes of the sensor network. Fortunately, generally, an exponentially increasing connectivity for the set of nodes in the sensor network is found as the number of nodes increases in the sensor network. The exponentially increasing connectivity ultimately leads to only one serially connected multipolling sequence as the number of nodes increases in the sensor network.

7. Conclusions

All MAC protocols proposed for wireless LANs can be categorized into contention-based or polling-based protocols and single or multi-hop protocols. In Wi-Fi sensor networks, where many sensors exist, contention-based protocols have a significant disadvantage because of the immense interference among sensors. Wi-Fi sensor networks may suffer from poor MAC performance. However, because of the inherent nature of polling-based protocols, they can avoid interference among sensors through efficient polling schemes. Single-hop protocols have become more attractive for Wi-Fi sensor networks because of recent sensor-technology advancements through which a great extension of the transmission range of sensors can be achieved. This study uses a polling-based single-hop MAC protocol based on a connectivity-based multipolling mechanism.

Since adopting the polling-based MAC protocol for supporting real-time traffic in wireless LANs, the connectivity-based multipolling MAC protocol has been researched to optimize the number of polling frame transmissions using the connectivity among Wi-Fi nodes. In particular, when Wi-Fi sensor networks connect a large number of Wi-Fi sensors to the Internet through wireless LANs, the optimization problem for minimizing the number of polling frame transmissions is challenging, and a real-time implementable algorithm for a minimal number of polling sequences is required for the successful implementation of the connectivity-based multipolling MAC protocol.

This paper proposes a real-time implementable, efficient node insertion algorithm for deriving a near-optimal number of polling sequences for the connectivity-based multipolling MAC protocol. The simulation results show that the number of polling sequences derived by the proposed node insertion algorithm converges to only one as the number of Wi-Fi sensor nodes increases in Wi-Fi sensor networks.

According to the proposed node insertion algorithm, each Wi-Fi sensor is granted an equal transmission opportunity to transmit its uplink data. However, some sensors may have urgent data, which should be transmitted with high priority. Future research should be conducted to modify the connected multipolling sequences to simultaneously satisfy the different service requirements of sensors.

Funding: This work was supported by the Dong-A University research fund.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Choi, W.-Y. Hybrid Polling Method for Direct Link Communication for IEEE 802.11 Wireless LANs. *EURASIP J. Wirel. Commun. Netw.* **2008**, *2008*, 598038. [CrossRef]
2. Choi, W.-Y. An Efficient Polling Scheme for Enhancing IEEE 802.11 PCF Protocol. *Frequenz* **2005**, *59*, 268–271. [CrossRef]
3. Choi, W.-Y. Combining Multipolling Method with Frame Aggregation for Collecting RFID Tag Information in IEEE 802.11 Wireless LANs. *AEU-Int. J. Electron. Commun.* **2011**, *65*, 345–348. [CrossRef]
4. Choi, W.-Y.; Chatterjee, M. Cluster-Based Multipolling Sequencing Algorithm for Collecting RFID Data in Wireless LANs. *Frequenz* **2015**, *69*, 141–147. [CrossRef]
5. *IEEE Std 802.11ah*; Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 2: Sub 1 GHz License Exempt Operation. IEEE Press: New York, NY, USA, 2017. [CrossRef]
6. *IEEE Std 802.11s*; Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 10: Mesh Networking. IEEE Press: New York, NY, USA, 2011. [CrossRef]
7. Singh, U.; Determe, J.-F.; Horlin, F.; Doncker, P.D. Crowd Forecasting Based on WiFi Sensors and LSTM Neural Networks. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 6121–6131. [CrossRef]
8. Bayrakdar, M.E. Priority Based Health Data Monitoring with IEEE 802.11af Technology in Wireless Medical Sensor Networks. *Med. Biol. Eng. Comput.* **2019**, *57*, 2757–2769. [CrossRef]
9. Alnazir, A.; Mokhtar, R.A.; Alhumyani, H.; Ali, E.S.; Saeed, R.A.; Abdel-khal, S. Quality of Services Based on Intelligent IoT WLAN MAC Protocol Dynamic Real-Time Applications in Smart Cities. *Comput. Intell. Neurosci.* **2021**, *2021*, 2287531. [CrossRef]
10. Choi, W.-Y. Energy-Efficient MAC Protocol for Wireless LANs with WiFi Sensors. *J. Electr. Eng.* **2021**, *72*, 352–355. [CrossRef]
11. Choi, W.-Y. Novel MAC Routing Protocol for Wireless Sensor Networks Based on IEEE 802.11 Ad Hoc Networks. *J. Electr. Eng.* **2022**, *73*, 355–358. [CrossRef]
12. Gupta, A.K.; Venkatesh, T.G. Design and Analysis of IEEE 802.11 Based Full Duplex WLAN MAC Protocol. *Comput. Netw.* **2022**, *210*, 108933. [CrossRef]
13. Cv, R.; Sathish, K. Performance Analysis of Clustered Based Underwater Wireless Sensor Network by Deploying Application as CBR. In Proceedings of the 2022 Third International Conference on Intelligent Computing Instrumentation and Control Technologies (ICICICT), Kannur, India, 11–12 August 2022. [CrossRef]
14. Jubair, A.M.; Hassan, R.; Aman, A.H.M.; Sallehudin, H.; Al-Mekhlafi, Z.G.; Mohammed, B.A.; Alsaffar, M.S. Optimization of Clustering in Wireless Sensor Networks: Techniques and Protocols. *Appl. Sci.* **2021**, *11*, 11448. [CrossRef]
15. Lokesh, D.; Reddy, N.V.U. Energy Efficient Routing Design for Target Tracking in Wireless Sensor Network. *WSEAS Trans. Inf. Sci. Appl.* **2022**, *19*, 132–137. [CrossRef]
16. Jassim, M.A.; Almobaideen, W.A. A Segment-Based Tree Traversal Algorithm for Enhancing Data Gathering in Wireless Sensor Networks. *WSEAS Trans. Comput.* **2021**, *20*, 66–73. [CrossRef]
17. Thamizhmaran, K.; Prabu, K. Performance Analysis of Hybridization of [PIO-GSO] Algorithms in Wireless Sensor Networks. *WSEAS Trans. Comput.* **2022**, *21*, 336–342. [CrossRef]
18. Romli, M.R.; Lee, J.-M.; Kim, D.-S. Hybrid MAC Protocol for UAV-Assisted Data Gathering in a Wireless Sensor Network. *Internet Things* **2021**, *14*, 100088. [CrossRef]
19. Dong, C.; Feng, S.; Yu, F. Performance Optimisation of Multichannel MAC in Large-Scale Wireless Sensor Network. *Int. J. Sens. Netw.* **2022**, *38*, 12–23. [CrossRef]
20. Darif, A.; Ouchitachen, H. Performance Improvement of a New MAC Protocol for Ultra Wide Band Wireless Sensor Networks. *J. Theor. Appl. Inf. Technol.* **2022**, *100*, 1015–1026.
21. Rehman, M.U.; Uddin, I.; Adnan, M.; Tariq, S.; Malik, S. VTA-SMAC: Variable Traffic-Adaptive Duty Cycled Sensor MAC Protocol to Enhance Overall QoS of S-MAC Protocol. *IEEE Access* **2021**, *9*, 33030–33040. [CrossRef]
22. Hai, T.; Zhou, J.; Padmavathy, T.V.; Md, A.Q.; Jawawi, D.N.A. Design and Validation of Lifetime Extension Low Latency MAC Protocol (LELLMAC) for Wireless Sensor Networks Using a Hybrid Algorithm. *Sustainability* **2022**, *14*, 15547. [CrossRef]
23. Yang, H.; Bao, L.; Luo, J.; Deng, B.; Wang, H. Utilizing Energy Efficiency and QoS Awareness for Receiver-Arbitrated and Sender-Predicted MAC Protocol in Wireless Sensor Network. *J. Phys.* **2021**, *1792*, 012071. [CrossRef]
24. Huamei, Q.; Linlin, F.; Zhengyi, Y.; Weiwei, Y.; Jia, W. An Energy-Efficient MAC Protocol Based on Receiver Initiation and Multi-Priority Backoff for Wireless Sensor Networks. *IET Commun.* **2021**, *15*, 2503–2512. [CrossRef]
25. Das, S.; Bhowmik, S.; Giri, C. Cross-Layer MAC Protocol for Semantic Wireless Sensor Network. *Wirel. Pers. Commun.* **2021**, *120*, 3135–3151. [CrossRef]
26. Alanazi, A.A. Comprehensive Analysis and Evaluation of Mobile S-MAC Protocol in Wireless Sensor Network. *Int. J. Comput. Sci. Netw. Secur.* **2022**, *22*, 358–366. [CrossRef]
27. Mathew, K.D.; Jones, A. CSRS-MAC: Cluster Based Synchronous Radio Scheduling MAC Protocol Using Carrier Sense Multiple Access for Wireless Sensor Network. *Wirel. Pers. Commun.* **2022**, *126*, 209–229. [CrossRef]
28. Goudjil, L.; Semchedine, F. COSFI-RIMAC: A Cooperative Short Frame Identifier Receiver Initiated MAC Protocol for Wireless Sensor Network. *Wirel. Pers. Commun.* **2023**, *128*, 2379–2395. [CrossRef]
29. Sun, N.; Wang, X.; Han, G.; Peng, Y.; Jiang, J. Collision-Free and Low Delay MAC Protocol Based on Multi-Level Quorum System in Underwater Wireless Sensor Networks. *Comput. Commun.* **2021**, *173*, 56–69. [CrossRef]

30. Latif, M.B.; Liu, F.; Liu, K. A TDMA-Based MAC Protocol for Mitigating Mobility-Caused Packet Collisions in Vehicular Ad Hoc Networks. *Sensors* **2022**, *22*, 643. [CrossRef] [PubMed]
31. Zhang, X.; Han, C.; Wang, X. Dual-Radio-Assisted (DRA) MAC Protocols for Distributed Terahertz Networks. *IEEE Open J. Veh. Technol.* **2021**, *2*, 111–124. [CrossRef]
32. Nilabar, N.U.; Manikandan, A.; Venkataramanan, C.; Dhanapal, R. A Score Based Link Delay Aware Routing Protocol to Improve Energy Optimization in Wireless Sensor Network. *J. Eng. Res.* **2023**, *in press*. [CrossRef]
33. Obaid, A.; Jaseemuddin, M.; Fernando, X. An Energy Harvesting MAC Protocol for Cognitive Wireless Sensor Networks. In Proceedings of the 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring), Helsinki, Finland, 25–28 April 2021. [CrossRef]
34. Obaid, A.; Fernando, X.; Jaseemuddin, M. A Mobility-Aware Cluster-Based MAC Protocol for Radio Frequency Energy Harvesting Cognitive Wireless Sensor Networks. *IET Wirel. Sens.* **2021**, *11*, 206–218. [CrossRef]
35. Bhar, J.; Bouazzi, I. Elastic Hybrid MAC Protocol for Wireless Sensor Networks. *Int. J. Electr. Comput. Eng.* **2021**, *11*, 4174–4182. [CrossRef]
36. Tolani, M.; Bajpai, A.; Balodi, A.; Sunny; Singh, R.K. Energy-Efficient Duty-Cycle Hybrid Medium Access Control Protocol for Wireless Sensor Network. *Wirel. Pers. Commun.* **2023**, *131*, 609–621. [CrossRef]
37. Roy, A.; Sarma, N. A Synchronous Duty-Cycled Reservation Based MAC Protocol for Underwater Wireless Sensor Networks. *Digit. Commun. Netw.* **2021**, *7*, 385–398. [CrossRef]
38. Dong, W.; Yang, Q.; Chen, Y.; Sun, S.; Huang, X. RHNE-MAC: Random Handshake MAC Protocol Based on Nash Equilibrium for Underwater Wireless Sensor Networks. *IEEE Sens. J.* **2021**, *21*, 21090–21098. [CrossRef]
39. *IEEE Std 802.11*; Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Press: New York, NY, USA, 2007. [CrossRef]
40. Zhong, R.Y.; Ge, W. Internet of Things Enabled Manufacturing: A Review. *Int. J. Agil. Syst. Manag.* **2018**, *11*, 126–154. [CrossRef]
41. Karaman, M.O.; Demir, S.; Börü, Ş.N.; Masat, S. Internet of Things: Success stores and challenges in manufacturing. In *Industry 4.0: Technologies, Applications, and Challenges*; Azizi, A., Barenji, R.V., Eds.; Springer: New York, NY, USA, 2023; pp. 57–92. [CrossRef]