Resource scheduling based on routing tree and detection matrix for Internet of things

Hongying Bai\textsuperscript{1,2}, Xiaotong Zhang\textsuperscript{1}, Yuxin Liu\textsuperscript{1} and Yingdong Xie\textsuperscript{2}

Abstract

Effective scheduling of limited communication resources is one of the critical methods for data transmission in the Internet of things. However, the time slot utilization rate of many existing resource scheduling methods of Internet of things is not high. This article proposes a new efficient resource scheduling based on routing tree and detection matrix for Internet of things. In heterogeneous Internet of things, according to the different working modes and functions, the nodes are divided into Internet of things devices, routing nodes, and base station. We use time slot multiplexing to improve the time slot utilization of continuous transmission in Internet of things. First, the time slot allocation table in a round is obtained by the time slot scheduling based on the routing tree. Then, the collision matrix and the transmission matrix are established based on the time slot allocation table in a round. Finally, the minimum time slot scheduling in continuous rounds is determined based on the routing tree and the detection matrix. The experimental results show that the resource scheduling based on routing tree and detection matrix effectively improves the utilization of time slots and improves the throughput of the Internet of things.

Keywords

Internet of things, resources scheduling, routing, matrix, time slot

Date received: 21 February 2021; accepted: 25 February 2021

Handling Editor: Yanjiao Chen

Introduction

With the development of 5G, more and more terminal devices and sensors are deployed in the Internet of things (IoT). It is estimated that the number of IoT connections will reach 4.1 billion in 2024, with an annual growth rate of 27\%.\textsuperscript{1} At present, the application of the IoT has penetrated many aspects of activities with the development of technology. IoT devices play a vital role in industry,\textsuperscript{2} autonomous vehicles,\textsuperscript{3} agriculture,\textsuperscript{4,5} healthcare,\textsuperscript{6} smart cities,\textsuperscript{7} and environment monitoring.\textsuperscript{8} The data transmission of the massive number of the sensor nodes is the main challenge in data collection of the IoT. Resource scheduling has great potential in ensuring IoT performance and avoiding transmission interference.\textsuperscript{9} Reducing the total time slot and improving the network throughput through reasonably use the limited resources in the communication network become one of the important research works in the IoT.

The resource scheduling methods generally need to compromise and optimize several designs because of differences of optimization objectives, interference models, and network environments. T Kim et al.\textsuperscript{10}

\textsuperscript{1}University of Science and Technology Beijing, Beijing, China
\textsuperscript{2}Ordos Institute of Technology, Ordos, China

Corresponding author:
Xiaotong Zhang, University of Science and Technology Beijing, No. 30 Xueyuan Road, Haidian District, Beijing 100083, China.
Email: zxt@ies.ustb.edu.cn

Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License (https://creativecommons.org/licenses/by/4.0/) which permits any use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (https://us.sagepub.com/en-us/nam/open-access-at-sage).
proposed a low-complex greedy algorithm to expedite the scheduling process which considers the node’s lifetime for deciding the active set of nodes. Kumar et al.\textsuperscript{11} reported resource scheduling during disaster situations. Priority-based stable matching algorithm is used for the allocation of resources for the corresponding activities.\textsuperscript{11} Wang et al.\textsuperscript{12} distributed a resource allocation solution for IoT which using an improved chaotic firefly algorithm that obtains the optimal location and working channel of secondary information gathering stations (SIGSs) to manage interference and resource allocation based on cognitive radio. Connectivity of nodes is critical for the IoT, as data collected need to be sent to the base station (BS). REN Moraes et al.\textsuperscript{13} proposed a link scheduling algorithm that minimizes the number of time slots needed to successfully schedule all the given links such that the nodes can communicate without interference in the signal to interference plus noise ratio (SINR) model. The various scheduling algorithm of IoT generally optimizes several designs because of differences of optimization objectives.

Resource scheduling usually aims to minimize wasting limited resources by efficiently allocating them among all nodes.\textsuperscript{14} In recent years, many research works on medium access control (MAC) protocol and resource allocation algorithms for the IoT have shown some results.\textsuperscript{15,16} Non-competitive scheduling is usually based on the time-division multiple access (TDMA) or the frequency-division multiple access (FDMA) or the code-division multiple access (CDMA) channel access mode to transfer data using a conflict-free manner. Yang and Wang\textsuperscript{17} investigated a dynamic allocation model for time and power resources in industrial IoT, reducing the energy loss of the communication system and ensuring the stability of communication between nodes. Liu et al.\textsuperscript{18} forward the optimal time scheduling of multiple modules which including spectrum sensing module, energy harvesting module, and ambient backscatter communication module (ABCom) by maximizing data transmission rate in IoT. Among them, the time slot allocation has become one of the main effective methods of resource scheduling.

Resource scheduling in a multi-hop environment poses a significant challenge due to the different resource requirement of each node. S Abdullah et al.\textsuperscript{19} proposed message scheduling with joint routing mechanism which is based on brokered architecture. The brokers choose scheduling strategy to transmit messages to the BS. P Gazori et al.\textsuperscript{20} focused saving time and cost on the scheduling of fog-based IoT applications using deep reinforcement learning approach. However, the few resource scheduling algorithms give consideration to both the time slot multiplexing and the time slot utilization rate.

This article proposes a new efficient resource scheduling based on routing tree and detection matrix (RSRM) for Internet of things. We use time slot multiplexing to improve the time slot utilization of continuous transmission in the IoT. The main contributions of this article are as follows. (1) We presented time slot scheduling based on the routing tree (TSRT). The TSRT obtains the time slot allocation table in a round. (2) New RSRM in IoT is presented which use time slot multiplexing to improve the time slot utilization of continuous transmission in IoT. (3) We conducted detailed simulation experiments to investigate the performance of the proposed TSRT and RSRM scheduling. The RSRM effectively improves the utilization of time slots and improves network throughput of IoT.

The rest of this article is organized as follows. Section “Related work” discusses the related work. Section “System model” gives the system model. In section “Proposed algorithm,” the RSRM in IoT is presented. We provide theoretical analysis in section “Theoretical analysis.” Section “Experiments” presents the experiments. Finally, the conclusions are given in section “Conclusion.”

Related work

In recent years, many researchers have addressed resource allocation in IoT. Samanta and Tang\textsuperscript{21} presented a dynamic micro-service scheduling (DYME) for the mobile edge computing in IoT platform and discussed the computational complexity of the scheduling algorithm in IoT. Olatinwo and Joubert\textsuperscript{22} presented novel approaches to the allocation of resources in Internet of things sensor network (IoTSN) systems applied to water-quality monitoring for optimization and more sustainable utilization of resources. IoT devices usually need to communicate with each other and some remote gateways through multi-hop communications. Qin et al.\textsuperscript{23} proposed a dual-interface dual-pipeline scheduling (DIPS) scheme that leverages low-power ZigBee interfaces to wake up a data pipeline constructed by high-power Wi-Fi interfaces fast pipelined data delivery in the IoT.

Wireless sensor network (WSN) is an essential component of IoT applications. To improve the utilization of network resources and reduce the delay of data transmission, some resource allocation schemes perform scheduling on channels and time slots. Gabale et al.\textsuperscript{24} indicated PIP (packets in pipe) algorithm which is suitable for multi-hop and multi-channel networks, especially for directional linear networks. Xu et al.\textsuperscript{25} proposed the use of slot reuse technology to improve the efficiency of resource scheduling in industrial WSNs. In this polling slot allocation, the network consists of subnetworks (FNs), backbone (BN).

Some researchers focus on collision-free resource allocations which are organized in a tree topology. Lee and Cho\textsuperscript{26} proposed tree-based time division multiple access...
MAC algorithm (tree TDMA) using time and frequency allocations. The tree TDMA supports full-duplex communication of voice and data. Osamy et al.\textsuperscript{27} proposed Effective TDMA scheduling for tree-based data collection using a genetic algorithm (ETDMA-GA). The genetic algorithm (GA) has been utilized to solve the generation of TDMA scheduling in ETDMA-GA.

Resource scheduling in a fair and efficient is major problem in multi-hop multi-channel networks. TDMA scheduling is extensively used for data delivery with the aim of minimizing the time slots for transporting data in multi-hop network.\textsuperscript{28} Xu et al.\textsuperscript{29} distributed a real-time scheduling in duty-cycled multi-hop sensor networks. The authors focused on periodic queries with sufficiently long time horizon in duty-cycled sensor networks in the presented scheduling. Multi-channel communication is an essential means to improve the efficiency of IoT. Tan et al.\textsuperscript{30} studied a multi-channel transmission scheme of a green IoT in underground mining and formulated it as a multi-channel multi-radio time slot co-scheduling problem. Zhang et al.\textsuperscript{31} proposed coloring route-tree based resource allocation algorithm for industrial WSNs. Gao et al.\textsuperscript{32} proposed an edge-based channel allocation (ECA) for unreliable IoT networks.

At present, a trend is the running different applications on heterogeneous sensor nodes deployed in network in order to better exploit the physical network infrastructure. Li et al.\textsuperscript{33} present the resource allocation in heterogeneous WSNs (SACHSEN) algorithm to make effective task-sensor assignments explicitly considering the performance requirements of application.

The implementation of data fusion plays a crucially important role in reducing the level of network traffic. F Alam et al.\textsuperscript{34} reviewed literatures on data fusion for IoT with a particular focus on mathematical methods and specific IoT environments. Jiang et al.\textsuperscript{35} proposed fairness-based packing of industrial IoT data in permissioned blockchains. The transaction packing algorithm not only achieves better fairness but also reduces the average response time as well.

Although the existing resource scheduling can satisfy the data transmission without conflict, but the utilization of time slot is not very high in the continuous transmission. In addition, many existing scheduling methods assume that nodes in network are homogeneous. This article proposes a new resource scheduling scheme in IoT (RSRM) where the nodes are divided into sensor nodes, routing nodes, and BS. The RSRM use time slot multiplexing to improve the time slot utilization of continuous transmission in IoT.

**System model**

This section mainly introduces the network model, the transmission interference model, and the symbol representation of RSRM in the IoT.

**Network model**

IoT is composed of a large number of nodes. According to the different working modes and functions, these nodes are divided into three types—IoT devices, routing nodes, and BS, as shown in Figure 1. The IoT devices collect and transmit data. The routing nodes are responsible for data fusion and routing. The BS collects data of all the nodes in each round.

In this article, the system model is based on the following assumptions:

- The position of nodes is fixed.
- The BS collects the data of all nodes.
- The routing nodes fuse, pack, and forward the received data.
- The length of the time slot is determined by the longest packet, which is adjustable.

Given an IoT network $G(V, E)$, where the $V$ represents a set of nodes in the network and the $E$ represents a set of links, $E \subseteq V \times V$. The set of available channels is $C = \{C_1, C_2, ..., C_k\}$. The optimization problem of resource scheduling can be described as the transmission of data to the BS in the minimum time slot and no conflict, which is shown as follows

$$
\min \left( \sum_{R=1}^{N} T_{SR} \right)
$$

s.t.

$$
C1 : f(u, v, T, C) \in \{0, 1\}, \forall u \in V, \forall v \in V
$$

$$
C2 : \sum_{v \in \text{Br}_u} f(u, v, T, C) \leq 1, \forall u \in V, \forall v \in V
$$

$$
C3 : \sum_{v \in \text{Br}_u} \left( f(u, v, T, C) + f(v, u, T, C) \right) \leq 1, \forall u \in V, \forall v \in V
$$

where the $T_{SR}$ represent the time slots in a round, the $R$ is the number of rounds, and the $\text{Br}_u$ represents the all neighbors of node $u$. In the constraint $C1, f(u, v, T, C) = 1$ indicates that at time $T$, node $u$ uses channel $C$.
to send data to node v, \( f(u, v, T, C) = 0 \) means that no operation is taken. The C2 indicates that in a time slot, a node \( u \) can only communicate with at most one neighbor node. The C3 indicates that a node only be either sending state or receiving state in a time slot.

**Transmission interference model**

Resource scheduling aims to transmit data while avoiding transmission conflicts through efficient resource allocation. This investigation is based on the graph-based protocol model, which includes a primary conflict and a secondary conflict. The primary conflict affects the range among one-hop neighbors, which occurs when one node attempts to do multiple operations at the same time as shown in Figure 2(a). The secondary conflict affects the range among the two-hop neighbors, which use the same channel to transmit. The secondary conflict is usually considered to be a hidden terminal problem, which is shown in Figure 2(b).

**Symbolic representation**

Some symbolic representation which is used in this article is shown in Table 1.

**Proposed algorithm**

In this section, the RSRM for IoT is presented. The structure of resource scheduling is shown in Figure 3. The nodes are divided into IoT devices, routing nodes, and BS. First, the time slot allocation table in a round is obtained by the scheduling based on the routing tree (TSRT). Then, the collision matrix and the transmission matrix are established based on the time slot allocation table. Finally, the minimum time slot scheduling in continuous rounds is determined based on the routing tree and the detection matrix.

The main idea of the RSRM is as follows:

- The time slot allocation table in a round \((SR(N_i, T_i))\) is obtained by the TSRT.
- The conflict matrix \((CM)\) is established based routing tree and topology of IoT.
- The transfer matrix \((TM)\) is established based on \(SR(N_i, T_i)\).

---

**Table 1.** Symbol representation.

| Symbol | Descriptions |
|--------|--------------|
| Num | The total number of nodes. |
| \(N_i\) | Node \(i\) in IoT. |
| \(C(N)\) | Channel of \(N\). |
| \(T_i\) | Time slot \(i\) in a round. \(T\) is the time slot table in a round. \(T = \{T_1, T_2, ..., T_{\text{max}}\}\). The \(T_{\text{max}}\) is the total number of time slots used in a round. |
| \(R\) | The number of continuous rounds. |
| \(SR(N_i, T_i)\) | \(SR(N_i, T_i)\) is time slot allocation table in a round. The \(N_i\) transmits data in slot \(T_i\) in a round. |
| \(CM\) | Conflict matrix \(CM\) derived by network topology and \(SR(N_i, T_i)\), \(CM_{ij} \in \{0, 1\}\). |
| \(TM\) | Transfer matrix \(TM\) derived by \(SR(N_i, T_i)\) and \(CM\), \(TM_{ij} \in \{0, 1\}\). |
| \(DM\) | Detection matrix \(DM\) is the matrix for detecting periodic overlap of slots in continuous rounds which is derived by \(CM\) and \(TM\). |
| \(U\) | The upper limit of non-overlapping time slots is obtained by matrix operation. The time slots can be reused after the \(U\) slot. |
| \(TS_i\) | \(TS_i\) is the time slot \(i\) in continuous rounds. \(TS\) is the time slot table in continuous round. \(TS = \{TS_1, TS_2, ..., TS_{\text{max}}\}\). The \(TS_{\text{max}}\) is the total number of slots used by the network in continuous rounds. |
| \(S(N_i, TS_i)\) | \(S(N_i, TS_i)\) is the final time slot allocation table. The \(N_i\) transmits data in slot \(TS_i\) in continuous rounds. |

---

**Figure 2.** Transmission interference model: (a) primary conflict and (b) secondary conflict.

**Figure 3.** Structure of resource scheduling based on routing tree and detection matrix.
The detection matrix ($DM$) is obtained according to the $CM$ and $TM$. The upper limit of the non-overlapping period ($U$) is calculated to ensure the normal execution of the time interval.

The minimum time slot allocation table in continuous rounds ($S(N_i, TR_i)$) is determined.

**TSRT**

The time slot allocation table in a round is obtained by the TSRT. The RSRM in the IoT is based on the TSRT. The routing tree is constructed according to the topology of the network. An example of a routing tree is shown in Figure 4. The nodes in IoT are divided into the IoT devices, routing nodes, and BS.

The channel allocation based on graph coloring is proposed in our previous work. The channel allocation based on graph coloring. The different colors represent the different channels. The BS chooses the channel $C_1$. The channels of routing nodes are calculated according to equation (2). The channel of the IoT device is consistent with their parent routing node, and the conflict is avoided by TDMA.

We presented the TSRT where each node is assigned a time slot in a round. The time slot allocation table in a round ($SR(N_i, T_i)$) is obtained by the TSRT which is shown in Algorithm 1.

The specific steps of TSRT are as follows:

**Algorithm 1.** Time slot scheduling based on routing tree (TSRT)

1. **INPUT:** Number of nodes ($Num$) and routing tree of IoT
2. **OUTPUT:** Time slot allocation table in a round $SR(N_i, T_i)$
3. $SR(N_i, T_i) \leftarrow \emptyset$
4. $T_i \leftarrow 1$
5. All the node are marked in non-traversed.
6. $Num' \leftarrow Num$
7. **while** ($Num' \neq 1$) **do**
8. **for** (non-traversed leaf node) **do**
9. Find the leaf node $N_i$ with the smallest ordinal number.
10. Time slot $T_i$ is allocated to $N_i$.
11. The information is added to the $SR(N_i, T_i)$.
12. Corresponding node is marked in traversed.
13. Leaf node $N_i$ is deleted from the topology.
14. $Num' \leftarrow Num' - 1$
15. **end for**
16. $T_i \leftarrow T_i + 1$
17. All the nodes are marked in non-traversed.
18. **end while**
19. The time slot allocation table of a round $SR(N_i, T_i)$ is obtained.
20. **return** $SR(N_i, T_i)$

- $S_i$: the routing tree is obtained according to the topology of network.
S2: the initial value of the time slot allocation table in a round is \( \Phi \). \( SR(\mathcal{N}_i, T_i) \leftarrow \{\Phi\}, T_i \leftarrow 1 \). All nodes are marked in non-traversed.

S3: we set a temporary number of nodes \( Num' \leftarrow Num \). The \( Num \) is the number of nodes in the network.

S4: if \( Num' = 1 \), we stop the current time slot scheduling process and skip to S8. Otherwise, S5 is executed.

S5: starting from the BS, depth traversal algorithm is use to find the leaf node \( \mathcal{N}_i \) with the smallest ordinal number on the non-traversed branches, and the time slot \( T_i \) is allocated to \( \mathcal{N}_i \). The information is added to \( SR(\mathcal{N}_i, T_i) \) and the corresponding branch is marked in traversed.

S6: the allocated leaf node \( \mathcal{N}_i \) is deleted from the temporary topology, and other nodes become leaf nodes again. We set \( Num' \leftarrow Num' - 1 \). If all the branches of the routing tree are traversed, skip to S7. Otherwise, skip to S5.

S7: the \( T_i \) is added to the time slot table \( T \) and \( T \leftarrow T_i + 1 \). \( T = \{T_1, T_2, ..., T_i\} \). All the nodes are marked in non-traversed, skip to S4.

S8: we obtain and store the time slot allocation table of a round \( SR(\mathcal{N}_i, T_i) \).

The process of time slot allocation in TSRT is shown in Figure 6. The adjacent nodes communicate with multi-channel, and the transmission of nodes does not interfere with each other. In Figure 6, The \( \mathcal{N}_1, \mathcal{N}_2, \mathcal{N}_4, \mathcal{N}_6, \) and \( \mathcal{N}_7 \) send data to their relay nodes in the first time slot \( (T_1) \). The \( \mathcal{N}_3, \mathcal{N}_5, \mathcal{N}_8, \mathcal{N}_{10}, \) and \( \mathcal{N}_{13} \) send data to their relay nodes in the second time slot \( (T_2) \). Until time slot \( T_7 \), all data are transmitted to the BS in a round. The time slot allocation table in a round \( SR(\mathcal{N}_i, T_i) \) is obtained as shown in Table 2. We get that all the branches of the routing tree are traversed, skip to S7. Otherwise, skip to S5.
the total number of time slots used in a round is \( T_{\text{max}} = 7 \) according to TSRT.

**RSRM**

The scheduling strategy of the above TSRT algorithm can ensure that all data are transmitted to the BS with the least number of slots in a single round. However, the performance of TSRT in continuous rounds is poor because each node needs to wait for a complete round before transmitting data to the other node.

We propose a new efficient RSRM to improve TSRT. We use the overlap of time slots to reduce the interval between two adjacent rounds, and improve the overall transmission efficiency of the network. The improved scheme aims to find the minimum time slot interval and improve the overall transmission efficiency of the network. The improved scheme aims to find the minimum time slot interval and improve the overall transmission efficiency of the network. The improved scheme aims to find the minimum time slot interval and improve the overall transmission efficiency of the network. The improved scheme aims to find the minimum time slot interval and improve the overall transmission efficiency of the network.

To determine whether a conflict exists between adjacent rounds, the \( CM \) is introduced. The \( CM \) is the \((Num - 1)\) order matrix, where the \( Num \) is the total number of nodes. The \( CM_{ij} \) represents that whether node \( i \) and node \( j \) interfere with each other in data transmission. The \( CM \) is defined as follows

\[
CM_{ij} = \begin{cases} 
1, & \text{If } N_i \text{ and } N_j \text{ interfere with each other} \\
0, & \text{Otherwise} 
\end{cases} \tag{3}
\]

The \( TM \) is introduced which records time slot allocation of nodes in a round. The \( TM \) is the \((Num - 1) \times T_{\text{max}}\) order matrix, where the \( T_{\text{max}} \) represents the total number of slots in a round in above \( SR(N_i,T_j) \). The \( TM_{ij} \) denotes that \( N_i \) transmits data to its parent node in \( T_j \). The definition of \( TM \) is as follows

\[
TM_{ij} = \begin{cases} 
1, & \text{If } N_i \text{ transmits data in slot } T_j \\
0, & \text{Otherwise} 
\end{cases} \tag{4}
\]

We introduce the \( DM \) to detect whether there is communication interference in time slot multiplexing during time slot allocation in continuous rounds. The \( DM \) is the \((Num - 1) \times T_{\text{max}}\) order matrix. As shown in equation (5), it indicates that there is communication interference among \( N_i \) and other nodes in the \( T_j \) when the value of \( DM_{ij} \) is not equal to 0 or 1. We define the time slot interval matrix \( TM' \) which moving \( TM \) some columns to the right. We set the initial value of \( TM' \) \( TM \). The new \( TM' \) deletes the rightmost column of the old \( TM' \) and splices the zero vectors with the same number of rows in the leftmost column. The \( DM \) is defined as equation (6)

\[
DM_{ij} = \begin{cases} 
(0|1), & \text{If } N_i \text{ interfere with other nodes in } T_j \\
0|1, & \text{Otherwise} 
\end{cases} \tag{5}
\]

\[
DM_{ij} = \sum_{k-i}^{j} CM_{ik} \times TM'_{ki} \quad \tag{6}
\]

The RSRM is based on the TSRT. We use the overlap of time slots to reduce the interval between two adjacent rounds. The final time slot allocation table \( S(N_i,TR_i) \) is obtained by the RSRM which is shown in Algorithm 2.

The specific steps of RSRM are as follows:

**Algorithm 2.** Resource scheduling based on routing tree and detection matrix (RSRM)

1. **INPUT:** Time slot allocation table in a round \( SR(N_i,T_j) \)
2. **OUTPUT:** The final time slot allocation table in continuous rounds \( S(N_i,TR_i) \)
3. \( \text{//Create the Conflict Matrix CM} \)
4. for \( i \leftarrow 0 \) do //The \( Num \) is the total number of nodes
5. for \( j \leftarrow 0 \) to \( Num \) do //The \( Num \) is the total number of nodes
6. if \( N_i \) and \( N_j \) interfere with each other then
7. \( CM_{i,j} \leftarrow 1 \)
8. else
9. \( CM_{i,j} \leftarrow 0 \)
10. end if
11. end for
12. end for
13. \( \text{//Create Transfer Matrix TM} \)
14. for \( i \leftarrow 0 \) to \( Num \) do
15. for \( j \leftarrow 0 \) to \( T_{\text{max}} \) do //The \( T_{\text{max}} \) is the max slot in a round
16. if \( N_i \) transmits data in \( T_j \) then
17. \( TM_{i,j} \leftarrow 1 \)
18. else
19. \( TM_{i,j} \leftarrow 0 \)
20. end if
21. end for
22. end for
23. \( \text{//Create the Detection Matrix DM} \)
24. \( TM' \leftarrow TM \) // Initializing interval matrix \( TM' \)
25. \( U \leftarrow 0 \) //Setting the initial value of \( U \)
26. \( DM_{ij} \leftarrow 2 \) ///Setting the initial value of \( DM_{ij} \)
27. while \( \text{max} (DM_{ij}) > 1 \) do
28. for \( i \leftarrow 0 \) to \( Num \) do
29. \( TM'[i][0] \leftarrow 0 \)
30. for \( j \leftarrow 0 \) to \( T_{\text{max}} \) do
31. \( TM'[i][j] \leftarrow TM'[i][j-1] \)
32. end for
33. end for
34. \( DM_{ij} = \sum_{k-i}^{j} CM_{ik} \times TM'_{ki} \)
35. \( U \leftarrow U + 1 \)
36. end while
37. The final time slot allocation table in continuous rounds \( S(N_i,TR_i) \) is obtained.
38. return \( S(N_i,TR_i) \)

- \( S_1 \): the TSRT algorithm has been executed, and the time slot allocation table in a round \( SR(N_i,T_j) \) is obtained by Algorithm 1.
- \( S_2 \): the \( CM \) is obtained by equation (3).
- \( S_3 \): the \( TM \) is obtained by equation (4).
- \( S_4 \): we define the time slot interval matrix \( TM' \) and set the initial value \( TM' \leftarrow TM \). The upper limit of the non-overlapping round \( U \) is recorded. We set the initial value \( U \leftarrow 0 \).
- \( DM \leftarrow \{2\} \).
S5: we delete the rightmost column of the matrix $TM'$ and splice the zero vectors with the same number of rows in the leftmost column.

S6: the detection matrix $DM$ is computed as equation (6).

S7: if the element of matrix $DM_{ij}$ is greater than 1, skip to S3. Otherwise, the RSRM algorithm is stopped. The $U$ is the required slot interval in continuous rounds. The $T_{\text{max}} - U$ time slots are reused between adjacent rounds.

S8: the final time slot allocation table in continuous rounds $S(N_t, T_R)$ is obtained according to the upper limit of non-overlapping time slots $U$.

Examples of RSRM

We take the routing tree and channel allocation above in section “TSRT” (Figure 5) as an example. The $CM$ is obtained as shown in Figure 7. According to TSRT and Table 2, the $TM$ is obtained by equation (4). The initial value of $TM'$ is $TM$. The new $TM'$ deletes the rightmost column of the $TM'$ and splices the zero vectors in the leftmost column.

The detection matrix $DM$ is calculated according to equation (6). As shown in Table 3, some elements of $DM_{ij}$ are greater than 1 when $U = \{1, 2, 3\}$. But, all values of $DM_{ij}$ are equal to 0 or 1 when $U = 4$, which indicates that there is no communication interference between nodes. Hence, $U = 4$ is the required time slot interval in continuous rounds. Time slots in continuous rounds can be reused after the $U$ time slots.

According to Table 2, in TSRT for six continuous rounds ($R = 6$), the total number of slots is 42 ($T_{\text{max}} = 7, TR_{\text{max}} = 7 \times 6 = 42$). In RSRM, we use the overlap of time slots to reduce the interval between two adjacent rounds and find the upper limit of non-overlapping time slots $U$. In the case, $U = 4$, which is shown in Table 3, we multiplex $T_{\text{max}} - U$ slots in each round. As shown in Table 4, we overlap 3 ($T_{\text{max}} - U = 7 - 4 = 3$) time slots in each round. The total number of time slots is 27 when $R = 6$ ($TR_{\text{max}} = 27$). Table 5 shows the final time slot allocation table. The RSRM saves 15 (42 - 27 = 15) time slots than TSRT for six continuous rounds. We assume that the unit time slot $\Delta t = 1$.

The less $U$, the more time slots saved in continuous rounds in RSRM. The $U$ is related to the topology and the routing tree of the network. The IoT devices switch to sleep state and reducing energy consumption and extending the lifetime of the whole network when they do not need to transmit or receive data.

Theoretical analysis

Theorem 1. The total time slot of the whole cycle of the network is $TR_{\text{max}}$. The total time slot of the network is $TR_{\text{max}}$, when $R$ tends to infinity

$$TR_{\text{max}} = (R \times U + T_{\text{max}} - U) \times \Delta t \tag{7}$$

$$TR_{\text{max}} = R \times U \times \Delta t \tag{8}$$

where the $T_{\text{max}}$ is the time slot in a round which is obtained by the TSRT. The $U$ is upper limit of the non-overlapping period which is obtained by the RSRM. The $\Delta t$ is length of unit time slot. The total cycle of the network has $R$ rounds.
Proof. In RSRM, we use the overlap of time slots to reduce the interval between two adjacent rounds, and improve the overall transmission efficiency of the network. The upper limit of the non-overlapping period ($U$) is the minimum time slot interval which is calculated by RSRM. Due to the use of optimization, $T_{\text{max}}^2$ time slots are reused between adjacent rounds. The total time slot of the whole life cycle is as follows

$$TR_{\text{max}} = R \times U \times \Delta t$$

The average slot time in per round ($T_{\text{arr}}$) is

$$T_{\text{arr}} = \frac{(R \times U + T_{\text{max}} - U) \times \Delta t}{R}$$

When the $R$ tends to infinity, the $T_{\text{arr}}$ is

$$\lim_{R \to \infty} T_{\text{arr}} = \lim_{R \to \infty} \left( \frac{(R \times U + T_{\text{max}} - U) \times \Delta t}{R} \right) = U \times \Delta t$$

Accordingly, the total time slot of the whole life cycle is $TR_{\text{max}} \approx R \times U \times \Delta t$ when $R$ tends to infinity.

Theorem 2. The throughput of the whole network is $TP$ when $R$ tends to infinity

$$TP \approx \frac{(\text{Num} - 1) \times d}{U \times \Delta t}$$

where the $d$ is the length of data transmitted by each node in a round. The $\text{Num}$ is the total number of IoT devices.

Proof. Throughput is defined as the amount of data transmitted in a unit time. In a single round, every IoT devices need to transmit a packet to the BS. Therefore, the throughput of the RSRM in one round ($TP_u$) is as follows

$$TP_u = \frac{(\text{Num} - 1) \times d}{T_{\text{max}} \times \Delta t}$$

For continuous multiple rounds, RSRM makes the rounds overlap and speeds up data transmission. The
throughput of the whole network in continuous $R$ rounds ($TP_R$) is as follows

$$TP_R = \frac{R \times (Num - 1) \times d}{\left[ R \times T_{\text{max}} - (R - 1) \times (T_{\text{max}} - U) \right] \times \Delta t}$$  \hspace{1cm} (14)$$

Take the limit of $R$ to get

$$\lim_{R \to \infty} TP_R = \lim_{R \to \infty} \left( \frac{R \times (Num - 1) \times d}{\left[ R \times T_{\text{max}} - (R - 1) \times (T_{\text{max}} - U) \right] \times \Delta t} \right)$$

$$= \lim_{R \to \infty} \left( \frac{R \times (Num - 1) \times d}{(R \times U + T_{\text{max}} - U) \times \Delta t} \right)$$

$$= \frac{(Num - 1) \times d}{U \times \Delta t}$$  \hspace{1cm} (15)$$

So, the throughput of the whole network is $TP \approx (Num - 1) \times d / U \times \Delta t$ when the round tends to infinity. By Theorems 1 and 2, we can see that the smaller upper limit of non-overlapping time slots ($U$), the less total time slot has been used, and the higher throughput will be.

**Experiments**

The nodes are distributed in the area $100 \text{ m} \times 100 \text{ m}$. The nodes of the network include IoT devices, routing nodes, and BS. The IoT devices are responsible for data collection and transmission. The routing nodes fuse and forward the received data. The BS collects the data of all nodes. The simulation parameters are given in Table 6.

The example of channel and time slot allocation in section “Experiments” in continuous three rounds which is according to RSRM is shown in Figure 8. The IoT devices are at the lowest level, and the BS is on the top. The different color represents the different channels and the $Ti$ represents the time slots. The IoT devices use the same channel as its relay node. The routing node jumps to the channel of the corresponding relay node before transmitting data.

Figure 9 shows the total time slots in continuous rounds in RSRM when the $U$ increases from 1 to 5 at different $T_{\text{max}}$ and rounds ($R$). The total time slot of the whole cycle of the network ($TR_{\text{max}}$) is obtained by the RSRM according to equation (7). The total number of time slots in a round ($T_{\text{max}}$) is obtained by the TSRT. The $U$ is the upper limit of the non-overlapping period which is related to the distribution of nodes, the routing tree of network, and the number of nodes. The total time slot of IoT increases with the value of the $T_{\text{max}}$, $U$, and $R$.

Figure 10 shows the average total time slots in continuous rounds used in RSRM when the number of nodes increases from 10 to 50, and the rounds increase from 1 to 8. Theoretically, the total number of time slots is related to the routing tree of network instead of the number of nodes. We take the average total time...
slots where $T_{\text{max}}$ and $U$ are relatively large when there are many nodes. The average total time slot of IoT increases with the number of nodes and rounds.

The RSRM uses the overlapping of time slots to reduce the interval between two adjacent rounds and improve the network’s overall transmission efficiency. The minimum slot allocation in continuous rounds is determined based on the matrix calculation to ensure the normal execution of the time internal. As shown in Figure 11, the total time slot of RSRM is significantly reduced compared with the TSRT.

Figure 12 shows the network throughput of the RSRM and TSRT. It indicates that the network throughput of RSRM is significantly improved compared with the TSRT. With the increase in network size, the network throughput increases.

Conclusion

A new RSRM in IoT is proposed in this article. The RSRM is based on the time slot allocation in a round is obtained by the TSRT. The $DM$ is obtained according to the collision matrix and the transmission matrix. The minimum time slot scheduling in continuous rounds is determined by RSRM which uses the overlapping of time slots to reduce the interval between two adjacent rounds in continuous rounds and improve the time slot utilization of IoT. Simulation results show that the RSRM scheduling algorithm effectively improves the utilization of time slots and improves network throughput of IoT. In further research, clustering-based scheduling algorithm is introduced when the number of nodes is large in IoT.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported, in part, by the National Natural Science Foundation of China (NSFC) project (no. 61671056) and the Scientific Research Projects of Ordos Institute of Technology (no. KYYB2018006).

ORCID iDs

Hongying Bai https://orcid.org/0000-0002-4345-7647
Xiaotong Zhang https://orcid.org/0000-0001-7600-7231

References

1. Vikhrova O, Pizzi S, Sinitsyn I, et al. An analytic approach for resource allocation of IoT multicast traffic.
In: Proceedings of the ACM MobiHoc workshop on pervasive systems in the IoT era, Catania, 2 July 2019, pp.25–30. New York: ACM.

2. Musengimana C and Yoo SE. Real-time scheduling based on multi-channel communication in IEEE 802.15.4 industrial Internet of things (IoT). In: Proceedings of the 2018 international conference on fuzzy theory and its applications (iFUZZY), Daegu, South Korea, 14–17 November 2018, pp.371–374. New York: IEEE.

3. Xiao Y, Krunz M and Shu T. Multi-operator network sharing for massive IoT. *IEEE Commun Mag* 2020; 57(4): 96–101.

4. Gupta N, Khosravy M, Patel N, et al. Economic data analytic AI technique on IoT edge devices for health monitoring of agriculture machines. *Appl Intell* 2020; 50: 3990–4016.

5. De Souza PSS, Rubin FP, Hohemberger R, et al. Detecting abnormal sensors via machine learning: an IoT farming WSN-based architecture case study. *Measurement* 2020; 164: 108042.

6. Gope P and Hwang T. BSN-Care: a secure IoT-based modern healthcare system using body sensor network. *IEEE Sens J* 2016; 16: 1368–1376.

7. El-Hosseini M, ZainEldin H, Arafat H, et al. A fire detection model based on power-aware scheduling for IoT-sensors in smart cities with partial coverage. *J Ambient Intell Hum Comput* 2021; 12: 2629–2648.

8. Kim D, Lee T, Kim S, et al. Adaptive packet scheduling in IoT environment based on Q-learning. *J Ambient Intell Hum Comput* 2020; 11: 2225–2235.

9. Kavitha K and Suseendran G. Priority based adaptive scheduling algorithm for IoT sensor systems. In: Proceedings of the 2019 international conference on automation, computational and technology management (ICACTM). London, 24–26 April 2019, pp.361–366. New York: IEEE.

10. Kim T, Qiao D and Choi W. Energy-efficient scheduling of Internet of things devices for environment monitoring applications. In: Proceedings of the 2018 IEEE international conference on communications (ICC), Kansas City, MO, 20–24 May 2018, pp.1–7. New York: IEEE.

11. Kumar JS, Zaveri MA and Choksi M. Activity based resource allocation in IoT for disaster management. In: Patel Z and Gupta S (eds) *Future Internet technologies and trends*, vol. 220 (Lecture notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering). Heidelberg: Springer, 2017, pp.215–224.

12. Wang Z, Liu D and Jolfaei A. Resource allocation solution for sensor networks using improved chaotic firefly algorithm in IoT environment. *Comput Commun* 2020; 156: 91–100.

13. Moraes REN, Dos Reis WWF, Rocha HRO, et al. Power-efficient and interference-free link scheduling algorithms for connected wireless sensor networks. *Wirel Netw* 2020; 26: 3099–3118.

14. Kim YG, Wang Y, Park BS, et al. A heuristic resource scheduling scheme in time-constrained networks. *Comput Electr Eng* 2016; 54: 1–15.

15. Danmanee T, Nakorn KN and Rojviboonchai K. CU-MAC: a duty-cycle MAC protocol for Internet of things in wireless sensor networks. *ECTI Trans Electr Electron Commun* 2018; 16(2): 30–43.

16. Lee G and Youn J. Group-based transmission scheduling scheme for building LoRa-based massive IoT. In: Proceedings of the 2020 international conference on artificial intelligence in information and communication (ICAIIC), Fukuoka, Japan, 19–21 February 2020, pp.583–586. New York: IEEE.

17. Yang O and Wang Y. Optimization of time and power resources allocation in communication systems under the industrial Internet of things. *IEEE Access* 2020; 8: 140392–140398.

18. Liu X, Gao Y and Hu F. Optimal time scheduling scheme for wireless powered ambient backscatter communications in IoT networks. *IEEE Internet Things* 2019; 6(2): 2264–2272.

19. Abdullah S, Asghar MN, Ashraf M, et al. An energy-efficient message scheduling algorithm with joint routing mechanism at network layer in Internet of things environment. *Wireless Pers Commun* 2020; 111(3): 1821–1835.

20. Gazori P, Rahbari D and Nickray M. Saving time and cost on the scheduling of fog-based IoT applications using deep reinforcement learning approach. *Future Gener Comp Sy* 2019; 110(10): 1098–1115.

21. Samanta A and Tang J. DYME: dynamic microservice scheduling in edge computing enabled IoT. *IEEE Internet Things* 2020; 7: 6164–6174.

22. Olatinwo SO and Joubert TH. Energy efficiency maximization in a wireless powered IoT sensor network for water quality monitoring. *Comput Netw* 2020; 176: 107237.

23. Qin H, Chen W, Cao B, et al. DIPS: dual-interface dual-pipeline scheduling for energy-efficient multihop communications in IoT. *IEEE Internet Things* 2019; 6(1): 718–733.

24. Gabale V, Chebrolu K, Raman B, et al. PIP: a multi-channel, TDMA-based MAC for efficient and scalable bulk transfer in sensor networks. *ACM T Sensor Network* 2012; 8(4): 28.

25. Xu H, Liu K and Guan X. Time slots allocating and multicycle scheduling in IWSN for narrow process automation. *Int J Distrib Sens N*, Epub ahead of print 18 June 2014. DOI: 10.1155/2014/324045.

26. Lee JH and Cho SH. Tree TDMA MAC algorithm using time and frequency slot allocations in tree-based WSNs. *Wireless Pers Commun* 2017; 95(3): 2575–2597.

27. Osamy W, El-Sawy AA and Khedr AM. Effective TDMA scheduling for tree-based data collection using genetic algorithm in wireless sensor networks. *Peer Peer Netw Appl* 2020; 13: 796–815.

28. Sgora A, Vergados DJ and Vergados DD. A survey of TDMA scheduling schemes in wireless multihop networks. *ACM Comput Surv* 2015; 47(3): 1–39.

29. Xu X, Zhao Y, Zhao D, et al. Distributed real-time data aggregation scheduling in duty-cycled multi-hop sensor networks. In: *Proceedings of the international conference on wireless algorithms, systems, and applications*, Honolulu, HI, 24–26 June 2019, pp.432–444. Cham: Springer.
30. Tan A, Wang S, Xin N, et al. A multi-channel transmission scheme in green Internet of things for underground mining safety warning. *IEEE Access* 2019; 8: 775–788.

31. Zhang X, Luo Q, Cheng L, et al. CRTRA: coloring route-tree based resource allocation algorithm for industrial wireless sensor networks. In: *Proceedings of the 2012 IEEE wireless communications and networking conference (WCNC)*, Paris, 1–4 April 2012, pp.1870–1875. New York: IEEE.

32. Gao W, Zhao Z, Yu Z, et al. Edge-computing-based channel allocation for deadline-driven IoT networks. *IEEE T Ind Inform* 2020; 16(10): 6693–6702.

33. Li W, Delicato FC, Pires PF, et al. Efficient allocation of resources in multiple heterogeneous wireless sensor networks. *J Parallel Dist Com* 2014; 74(1): 1777–1788.

34. Alam F, Mehmood R, Katib I, et al. Data fusion and IoT for smart ubiquitous environments: a survey. *IEEE Access* 2017; 5: 9533–9554.

35. Jiang S, Cao J, Wu H, et al. Fairness-based packing of industrial IoT data in permissioned blockchains. *IEEE T Ind Inform*. Epub ahead of print 21 December 2020. DOI: 10.1109/TII.2020.3046129.