A Minimized Buffered Link Scheduling Algorithm for WirelessHART Networks Based on Graph Route

Nanhang Luo¹,a, Mingjie Fang¹, Kunming Zhao³, and Enwu Du⁴
¹Wuhan Second Ship Design and Research Institute, Wuhan 430064, China.
²redstarzhang@futunn.com
³Corresponding author’s e-mail: luonh1101@163.com

Abstract. WirelessHART network is an emerging wireless sensor network technology, famous for simplicity, safety, and reliability. WirelessHART network uses Time Division Media Access (TDMA) and Frequency Hopping Spread Spectrum (FHSS) access mechanism to avoid collisions and mitigate interference from other wireless networks. Many formulations of the link scheduling problem have been proven to be NP-complete, which is about how to assign slots and channels to links. In this paper, we propose a minimized buffered link scheduling algorithm for WirelessHART networks based on graph route (LBLSGR), which utilizes edge coloring to scheduling time slots according to the resource demand of nodes. We supply more communication resource to the nodes located nearer to the gateway, because of frequent packet forwarding. Utilizing this algorithm, the WirelessHART networks can make the average network delay least, the average network power assumption lower, and the packet buffers minimized. At last, the validity of our algorithm is proved and the test result is provided.

1.Introduction
A WSN network can be used in many scenarios such as household appliances control, equipment monitoring, asset managing, as well as process control. Most of wireless sensor network use TDMA or CSMA access mechanism to arbitrate and coordinate communication between network devices, in which time is slotted and every data transaction should be achieved during a slot, so the rational link scheduling is crucial to make sure the network operate normally. Link scheduling is an open issue about how to allocate time slots and channels to nodes, which will affect the communication quality, throughput, real-time, power consumption and reliability of the whole network. So far, the method of link scheduling is divided into distributed link scheduling and centralized link scheduling. For distributed link scheduling algorithm, the nodes and their neighbors determine the business data transmission time slot and channel through the handshake negotiation and other mechanisms. The throughput and link utilization is low because this algorithm does not focus on the entire network. Centralized link scheduling algorithm utilizes the center node to configure and schedule network resources, which aims at the best performance of the whole network. The disadvantage is that it may increase the communication overhead during the allocation of time slots and channels.

The issue that assign each link with corresponding time slot and channel can be treated as coloring the network graph [1]. Vizing’s theorem [2] stated that a valid edge coloring for a graph can be obtained by using at most \( \Delta + 1 \) colors, where \( \Delta \) was the maximum node degree in the graph, and the algorithm was given by J. Misra and David Gries [2], but it does not make full use of the channels. Zhang Xuedan proposed a distributed link scheduling algorithm based on parallel graph coloring, which allocates...
communication resources relying on the local network information [3]. The algorithm could alleviate implicit conflicts but not absolutely avoid them. Zhao Jindong used a vertex coloring algorithm based on breath-first search to assign slots and channels, making the length of superframe least [4]. The disadvantage is that it does allocate more slots to the nodes nearer to the gateway, which always forward packets from other nodes. So, the network delay of this algorithm is significant and the ratio of link utilization is low. Haibo Zhang demonstrated that the minimum time to complete convergence in networks with multi-line tree routing topology is \( \max\{2n_1-1, N\} \) time slots where \( N \) is the number of nodes in the network and \( n_1 \) is the maximum number of nodes in a line sub-tree[5]. The advantage of this solution is that it decreases the network delay and increase the ratio of link utilization. But it did not solve the problem with link scheduling based on mesh topology.

The remainder of this paper is organized as follows. Section 2 describes the wirelessHART protocol and its link scheduling. The wirelessHART link scheduling is modeled in Section 3. In Section 4, we will design our link scheduling algorithm for wirelessHART networks. Finally, remarks and future work are given in Section 5.

2. Background

2.1. WirelessHART Overview

WirelessHART is one kind of field bus, originating from Highway Addressable Remote Transducer (HART). Officially released in September 2007 by HCF, it is the first open and interoperable wireless communication standard for process automation. WirelessHART is a complete wireless mesh networking protocol mostly based on the IEEE 802.15.4-2006 DSSS physical layer standard, operating in the license-free ISM 2.4GHz band with a data rate of up to 250 kbps. The data link layer defines a strict 10ms timeslot and utilizes TDMA and FHSS technology to provide collision-free and deterministic communications. The network layer employs mesh topology, and support graph routing mechanism to provide redundant paths which allows messages to be routed around physical obstacles, broken links, and interference. The transport layer establishes secure bi-direction transport pipe which can improve the end-to-end data transmission reliability [6]. Fig 1 shows a typical wirelessHART network.

2.2. WirelessHART Link Scheduling

In WirelessHART protocol communications are precisely scheduled using an approach referred to as Time Division Multiple Access (TDMA). The vast majority of communications are directed along graph routes. Scheduling is performed by a centralized network manager which uses overall network routing information in combination with communication requirements that devices and applications have provided. The schedule is subdivided into slots and transferred from the network manager to individual devices; devices are only provided with the slots for which they have communication requirements. The network manager continuously adapts the overall network graph and network schedule to changes in network topology and communication demand. WirelessHART networks support time and frequency diversity, but don’t allow spatial diversity. So, we can’t assign two nodes-pair to communicate on the same channel and at the same time slot even if they are located out of the communication range of each

![Fig 1. WirelessHART network][5]
other. The data flow of WirelessHART networks is always from field devices to the gateway or from the gateway to field devices, so we study the situation that the packets converge to the gateway.

As far as we know, we can schedule slots and channels utilizing coloring theory to avoid data conflicts. So, we can improve typical edge coloring algorithm to serve the wirelessHART network and propose a minimized buffered link scheduling algorithm based on graph route (LPTSGR).

3. Scheduling model

3.1. Conflict Model

In this paper, we study the nodes which are half-duplex single-transceiver, so they can’t send and receive data at the same time. The communication distance between two nodes is limited. The conflict exists between two nodes, as long as the ratio of signal to noise meets a certain threshold. Formula (1) depicts the ratio of signal to noise for node i and node k. According to the causes of conflict, we divide the types of conflict into explicit conflict and implicit conflict.

\[ \text{SNR}(i,k) = \frac{P_i}{L_{n(i,j)} N_i} \geq \text{snr}_b \] (1)

There are two kinds of conflicts in wireless multi-hop mesh networks: explicit conflicts and implicit conflicts [1]. The situation that a vertex is in more than one link at the same time is explicit conflict, such as link A->B and link C->B that shown in Fig.2 (a); This type of conflict can be avoided by assigning each link to different time slots. The wireless interference from the other links at the same channel in neighborhoods is implicit conflict. In Fig.2 (b), when A->B and C->D send data at the same channel simultaneously, the implicit conflict will be engendered [4]. We can allocate each link to different channel and the same slot to avoid the implicit conflict and increase the ratio of link utilization.

![Fig 2. Explicit and Implicit conflict](image)

3.2. Network Model

We can model the WirelessHART network as an acyclic simple graph where the vertices in represent the network devices and the edges in denote the device pairs that can sustain reliable communication. The superframe is defined as set, where is the time slot of superframe and its number is. We schedule slots and channels using graph coloring theory, and the color represents the time slot, so the length of the superframe is equal to the number of colors. Our optimization goal is to minimize the number of colors, that is, to make superframe shortest. At the same time, we should allow nodes to buffer the least packets to save the hardware RAM resource. In order to achieve our goals, the network model must comply with some constraints as follows:

1. All packets should converge to the gateway.
2. One node can’t send and receive the packet at the same time.
3. At any slot, the number of packets should be constant.
4. At one slot, a node can’t receive packets from different nodes.
5. Each node can buffer at most one packet.
6. Explicit conflict and implicit conflict should be avoided.
7. The slots allocated to a node should include sending its packet and forwarding other packets.
4. Lmlsg algorithm

4.1. Optimize Network Topology

In wirelessHART networks, each data is converged to the gateway along the mesh networks utilizing the graph route, which is a collection of paths that connect network nodes. Every node gets the next-hop destination by finding out the entries of graph 255, which indicates the shortest path to the gateway. In order to decrease the network delay accumulated with hops, we optimize the network topology according to graph route and establish a hierarchical structure. We delete the link between the brother nodes, preserving the links to father node and uncle nodes, so we can conduct a new topology based on minimum hops. Figure 3 shows how the optimized topology is created. We appoint the gateway as the center node with its level 0. The children of the gateway are designated to the level 1, whose children are designated to the level 2, and so on. Figure 4 is the flow chart of creating an optimized topology.

![Original topology](image1)

![Optimized topology](image2)

Fig 3. Optimize the network topology

![Flow chart](image3)

Fig 4. Flow of conducting a clock topology
4.2. Edge coloring

In order to avoid the communication conflict, edges with a common vertex should not be colored the same color. Edges without a common vertex can be colored the same color, but must be assigned to different channels. We use the least number of colors to color all the edges of the graph, making the network delay least. Before coloring, we suppose that each node product a packet periodically, these packets converge to the gateway at last. When coloring the edges, some principles should be adhered to as follows:

1. Repeated coloring is allowed, and we should color the edges nearer from the gateway first.
2. The edge can be colored only when there is a packet buffered in the starting point and no packet buffered in the end point of the edge.
3. The node has a higher coloring priority, whose ID is smaller.
4. Each edge should be colored at least once, making sure the link redundancy and increasing the network reliability.

The gateway can receive only one packet in a slot, so at least N slots are needed to receive N packets. When the node whose depth is one sends a packet to the gateway at slot \( t \), its child node should send a packet to it at slot \( t+1 \) if there is a packet buffered in the child node. All the nodes who send packets at slot \( t \) can receive packets at slot \( t+1 \), and receive the packets from the nodes who receive packet at slot \( t \). In figure 3(b), if link 2->GW, 3->1, 7->4, 9->6 are colored at slot \( t \), nodes 3,7,9 and the gateway should receive data from node 1,4,6. The process of edge coloring algorithm is as follows:

1. Initialize the color ID \( slot = 0 \), and the network depth information \( depth = 0 \).
2. \( slot = slot + 1 \), \( depth = depth + 1 \), if \( depth = MaxDepth \) go to (5), else go to (3).
3. Color the edges between the nodes whose depth is \( depth \) and the nodes whose depth is \( depth - 1 \).
4. \( depth = depth + 1 \), if all the packet converge to the gateway, go to (6), else go to (3).
5. \( slot = slot + 1 \), if all the packet converge to the gateway, go to (6), else go to (3).
6. End coloring, and assign the nodes to different channels, which are allocated to the same slots.

We use the hardware platform of msp430 and cc1100e to verify our LMLSGR algorithm, make some comparison with NRS algorithm [4]. Figure 6 shows that LMLSGR needs fewer slots than NRS to schedule, and the ratio of link utilization of the former is higher than the latter, so the throughput of LMLSGR is higher than that of NRS. Figure 7 depicts that the delay of each packet in LMLSGR less than that in NRS, and the average network delay is less. We can know form figure 8 that each node needs only one packet buffer in LMLSGR, but more than one in NRS, so our algorithm can save more hardware RAM resource. Figure 9 indicates that each node needs less power consumption in LMLSGR than that in NRS, so the average network power consumption of the former is less than that of the latter.
5. Conclusion
Time synchronization is a crucial task in wirelessHART networks because of slotted communication mechanism. The link scheduling is a crucial issue which affects the network delay, throughput, power consumption and reliability. In this paper, we propose an innovative algorithm (LMLSGR) to decrease the network delay, and save the RAM resource. Our algorithm is based on the edge coloring, but assigns more communication resources to the nodes which are nearer to the gateway. We give an example of 11 nodes, and utilize the msp430 and cc1100e to verify the algorithm. From the comparison with NRS, we can see clearly that LMLSGR can increase network delay, save power consumption and RAM resource.
References

[1] D. A. Grable and A. Panconesi, “Nearly optimal distributed edge colouring in O(log log n) rounds,” Proc. ACM/SIAM Symposium on Discrete Algorithms, pp.278–285, 1997.

[2] J. Misra, David Gries, A constructive proof of Vizing’s Theorem, Information Processing Letters, v.41 n.3, p.131-133, March 6, 1992.

[3] Xuedan Zhang, Jun Hong, Lin Zhang, Xiuming Shan, Victor O.K. Li, CC-TDMA: Coloring-and Coding-based Multi-channel TDMA Scheduling for Wireless Ad Hoc Networks, Proc. of IEEE WCNC 2007[C], Hong Kong, China, March, 2007. p.133–137.

[4] Zhao Jindong, Wang Qin, Yu Feng, “A Route-Based Scheduling in Wireless Multi-Hop Mesh Networks for Collision Avoidance”, Wireless Communications, Networking and Mobile Computing. 2009,p:1-4

[5] H. Zhang, P. Soldati, and M. Johansson, “Optimal link scheduling and channel assignment for convergecast in linear WirelessHART networks,” in IEEE WiOpt, Seoul, Korea, Jun 2009

[6] Industrial communication networks – Wireless communication network and communication profiles –WirelessHART™ IEC 62591.