Phase Switching Mechanism for WiFi-based Long Distance Networks in Industrial Real-Time Applications

Jintao Wang[1,2], Xi Jin[1], Peng Zeng[1], Zhaowei Wang[1,2] and Ming Wan[2]
[1] Lab. of Industrial Control Network and Systems, Shenyang Institute of Automation Chinese Academy of Sciences,
Shenyang, China, 110016
[e-mail: zp@sia.cn]
[2] University of Chinese Academy of Sciences,
Beijing, China, 100049
[e-mail: wangjintao@sia.cn]
*Corresponding author: Peng Zeng

Received June 6, 2016; revised October 13, 2016; accepted November 18, 2016;
published January 31, 2017

Abstract

High-quality industrial control is critical to ensuring production quality, reducing production costs, improving management levels and stabilizing equipment and long-term operations. WiFi-based Long Distance (WiLD) networks have been used as remote industrial control networks. Real-time performance is essential to industrial control. However, the original mechanism of WiLD networks does not minimize end-to-end delay and restricts improvement of real-time performance. In this paper, we propose two algorithms to obtain the transmitting/receiving phase cycle length for each node such that real time constraints can be satisfied and phase switching overhead can be minimized. The first algorithm is based on the branch and bound method, which identifies an optimal solution. The second is a fast heuristic algorithm. The experimental results show that the execution time of the algorithm based on branch and bound is less than that of the heuristic algorithm when the network is complex and that the performance of the heuristic algorithm is close to the optimal solution.

Keywords: Industrial control network, WiFi-based long-distance network, real-time traffic, time delay, 2P-MAC

This research was supported by the National Nature Science Foundation of China (No. 61502474，61233007, 61501447).

https://doi.org/10.3837/tiis.2017.01.005 **ISSN : 1976-7277**

1. Introduction

The industrial monitoring and control network is a recently developed network technology in the field of automatic control [1]. It is used in combination with computer networks, communication technology and automatic control technology. Due to its advantages of low cost, high reliability, easy maintenance, and flexibility of wireless technology, the industrial monitoring and control network has increasingly been applied to industrial monitoring and control fields.

The data transmitted in industrial monitoring network contain two main types of traffic in industry fields. The first type is the real-time management or control information [2]; the other is the graphics or image information that is not delivered in real time but incurs high bandwidth demand. In this paper, the "real-time" term refers to the traffic that has real-time demand. In industry field, the real-time is an important indicator. In that case, the wireless network needs to meet the different traffic transmission demands in real time or at high bandwidth. At the same time, in the context of economic globalization and smart plants, information interacts between plants and the different industrial parks of the same plant more frequently. However, plants and industrial parks of the same plant are sometimes far apart, as in facilities for oil and gas production, and the distance between oil well sites, stations and libraries can range from a few miles to a tens of kilometers [3]. Therefore, the demands of long-distance transmission should be considered.

The classic IEEE 802.11 protocol can provide high bandwidth, but its MAC protocol [4] is based on Carrier Sense Multiple Access (CSMA). The CSMA protocol makes the transmission delay unpredictable and cannot ensure the system's real-time performance [5]. This feature restricts widespread use of the IEEE 802.11 protocol in industrial systems. By contrast, WiFi-based Long Distance (WiLD) networks with IEEE 802.11 protocols can support predictable transmission delays.

An IEEE 802.11 WiLD network [6] refers to a multi-hop wireless WiFi network in which the distance between adjacent nodes is very long (dozens to hundreds of kilometers). These nodes use high-power IEEE 802.11 wireless cards and high-gain directional antennas to realize long-distance traffic transmission. Using public industrial, scientific and medical (ISM) frequencies and clean energy such as solar as its power does not rely on the mains supply and can work in harsh outdoor environments.

In [7], Professor Raman from the Indian Institutes of Technology (IITs) noted that the performance of the IEEE 802.11 CSMA/CA protocol is very poor in long-distance transmission. He proposed the 2P-MAC protocol, the network architecture and the interference model of the WiLD network. Since then, much research has been conducted using the 2P-MAC protocol. Meanwhile, the 2P-MAC protocol has become the most common MAC protocol in the field of WiLD networks. Professor Brewer's team systematically studied the WiLD network [8], including the MAC protocol, network deployment, and network management. In addition, experimental networks have been established in India, Ghana, Diego bay and other places. International companies such as Intel, IBM, Microsoft have also conducted research and productization based on the IEEE 802.11 protocol, such as the RCP (Rural Connectivity Platform) project that Intel published [9].

Some researches have been published in recent years. Protocols based on the 2P-MAC protocol for WiLD networks such as WiLDNet [8], JazzyMAC [6] and JaldiMAC [10] have been designed. Zhao et al. [11] proposed a traffic scheduling protocol based on 2P-MAC protocol that provides a cross layer admission control mechanism. A QoS aware multipath routing protocol for WiLD networks is proposed in [12]. [13] proposed an anticipatory packet scheduling protocol based on flow that prepares a fine-tuned transmission schedule according to the QoS demands of different traffic priority classes. An energy efficient rate adaptation algorithm was proposed in [5] for WiLD links. It adopts an online measurement of FDR-RSSI (Frame Delivery Ratio-Received Signal Strength Indication) mapping to choose the bit rate based on the joint considerations of RSSI and transmission power. However, the technology described in the aforementioned studies was not designed for real-time industrial traffics.

In the 2P-MAC protocol, when a packet is transmitted along a routing path, its delay is fixed and equal to the multiplication of the number of hops by the time duration of each hop (the detailed description is shown in Section 2). In a long distance network with 2P-MAC protocol, the transmission delay can be predicable and we can determine whether the delay satisfies the system real-time performance. If the fixed delay cannot satisfy the real time demands, it can be met by changing the number of time slots in transmitting and receiving phase in some nodes. However, we do not know how many time slots and which nodes we have to change. This is the problem we focus on in this paper.

Therefore, we propose some methods to determine the number of time slots contains in the transmitting and receiving phase of each node to improve the real-time performance and reduce the switching overhead. First, we introduce our focused problem and formulate it as a maximizing problem. Second, we propose an algorithm called **PCL-BAB** (a method to solve Phase Cycle Length based on Branch And Bound algorithm) to solve the problem. The optimum solution can be obtained by this algorithm. To further reduce the computational complexity, a heuristic algorithm called **PCL-H** (a Heuristic method to solve Phase Cycle Length) is proposed. Third, we simulate the algorithms with different network and number of traffic flows, the experimental results show that with PCL-BAB we can obtain the optimal results and that the heuristic algorithm PCL-H can significantly reduce the search time compared with the traditional traversal search method.

The WiLD nodes are powered by the grid or solar, and the power consumption of the WiLD base station is relatively lower (about 5W) [14]. In that case, an accumulator with the capacity 12V-100Ah can supply energy continuously for a WiLD base station up to 238 hours without charging. Moreover, the accumulator can be solar-charged. Thus, we assume that the power is sufficient in WiLD network. Besides, the algorithms we proposed is computed offline and the data transmission mechanism of nodes is not changed. Therefore, these algorithms do not bring extra power consumption to the network. Thus, in this paper, we do not consider the power consumption problem.

2. Problem Descriptions

2.1 2P-MAC Protocol

IEEE 802.11 protocol was originally designed for an indoor, short-distance, broadcasting local area network (LAN) [15]. When it is used for long-distance transmission, there are problems such as the low-efficiency link layer retransmission mechanism and conflict and interference between links. Aiming to resolve such problems, Raman proposes a low-cost solution that involves modifying an 802.11 network card to realize SynTx and SynRx in different time slots. The phase transition diagram in **Fig. 1** illustrates the 2P operation in the process of phase switching. Each network node has two phases: SynTx and SynRx. The duration of each phase is also fixed. During the SynTx (SynRx) phase, all network interfaces in a node must transmit (receive) simultaneously. All nodes switch their phases from one to another at the fixed time. In each routing path from data source node to the destination, any two adjacent nodes cannot work in the same phase; i.e., when a node sends a packet, the next node can receive it. Thus, the duration of each hop is fixed. The phase assignment of all nodes can be obtained via a binary-graph [16].

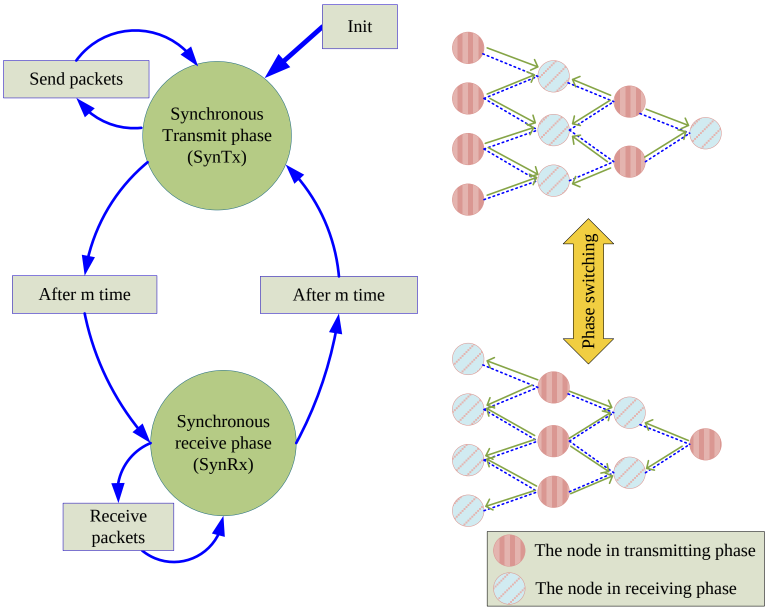
Init
Send packets
Synchronous Transmit phase (SynTx)
After m time
After m time
Synchronous receive phase (SynRx)
Receive packets
Phase switching
The node in transmitting phase
The node in receiving phase

Fig. 1. Phase Switching with the 2P-MAC Protocol

2.2 Problems in Phase Switching

In the above 2P operation process, there is a problem that the topology must be a binary graph. When some nodes are in phase SynTx, the nodes connected to them (link-nbr) must be in SynRx phases and vice versa. In other words, in the operation of 2P, the topological adjacency nodes must be in different phases.

To solve this problem, JazzyMAC[6] allows each link to dynamically adapt the length of its transmission slots based on the locally observed traffic load using a token in each link, making the protocol applicable to arbitrary topologies. However, the key limitation of JazzyMAC is that a node can transmit only when it has the token of all its links. In each slot, a large number of idle nodes wait for tokens. Therefore, the authors of [17] propose Algo-1 and an extended version Algo-2 [18] , Algo-d [19] to minimize the schedule's superframe length by solving the MAX-CUT problem.

However, there is another problem: switching has overhead, whereas the WiLD network uses a transmitting and receiving switch to realize the information transmission. According to [7], When the bandwidth is 11 Mbps in IEEE 802.11, if the durations of both transmitting and receiving are 6.5 ms, the overhead caused by switching account for 11% of the bandwidth (the switch time delay is approximately 0.72 ms). However, if a shorter end-to-end delay is desired, the length of the transmitting and receiving phase cycle must be reduced appropriately. When the transmitting and receiving phase cycle length is smaller, the proportion of switching overhead and the loss of bandwidth are greater.

Therefore, based on the real time demands of traffics in the network, this paper aims to minimize the switching overhead. Namely, under the condition of guaranteed time delay, let each phase cycle be as long as possible. After solving, we can obtain the phase cycle length of each node that can meet the real time demands and maximally the phase utilization.

The research described in this paper is based on the heterogeneous backhaul network, as shown in Fig. 2. It encompasses various network forms, including WirelessHART, Zigbee and WIA-PA. These subnets access to the data center through IEEE 802.11 WiLD network. The data are also stored and analyzed in the data center.

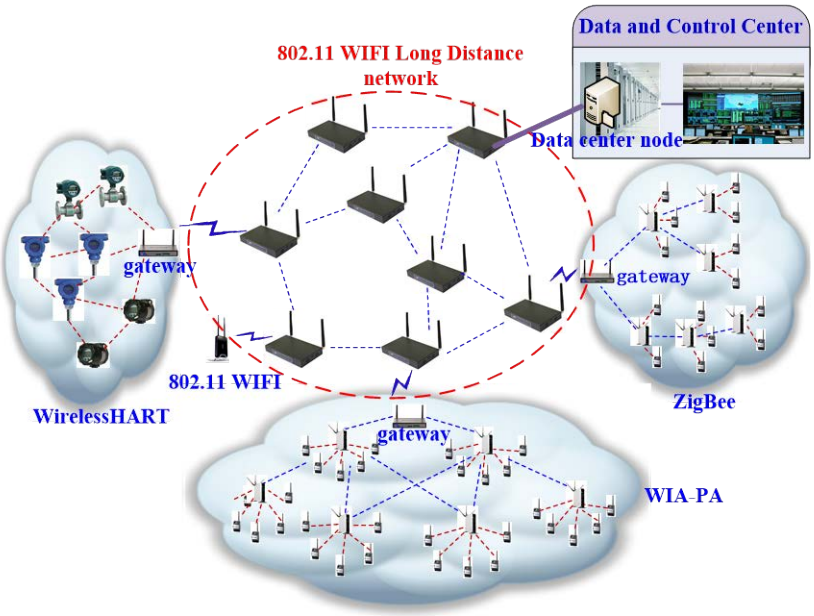

Fig. 2. Application scenario of the IEEE 802.11 WiFi-based long distance network

3. Problem Modeling

We model a multi-hop WiLD Mesh network as an arbitrary graph $G(V,E)$, where V denotes the set of nodes and E represents the set of directed links. Let $e(i,j)\in E$ denotes a directed link from node n_i to n_j. We write $L_{e(i,j)}\in[0,1]$ as the load of link $e(i,j)$, which is fixed for each link. Let L denotes a column vector with $|E|$ elements that records the load of all links,i.e $\mathbf{L}=\{L_{e(i,j)}\,|\,e(i,j)\in E\}$.

Definition 1. Node phase cycle. The node phase cycle refers to a successive transmitting phase and receiving phase along with two switching delay of the nodes that use the transmitting and receiving phase switching mechanism in a WiLD network. In this paper, we assume the length of transmitting phase and receiving phase is the same.

Definition 2. Phase cycle length. The phase cycle length refers to the number of time slots that a node phase cycle contains in a WiLD network using transmitting and receiving phase switching mechanism.

WiLD networks use transmitting and receiving switching mechanisms to realize the information transmission, but switching has overhead. So we have the following definition.

Definition 3. Switching overhead. The switching overhead is the time delay incurred during phase switching. Because of the limitations of the software and hardware performance of the network devices, phase switching requires a certain amount of time. The delays caused by the performance of the software and hardware in phase switching are constant and are denoted by τ_c, the unit of which is the unit time slot.

we assume that the length of a phase cycle length is T_i. The larger the value of T_i, the smaller the effect of the switching overhead in each cycle. However, a large value of T_i means a longer delay because in the worst case, after arriving at a node, the packet may need to wait for a full cycle length before it can be transmitted. The longer the cycle length, the longer the waiting time for the packet. We use the phase variable s_{it} (or r_{it}) to denote a node in transmitting (or receiving) phase. s_{it} and r_{it} are defined as follows:

$$s_{it} = \begin{cases} 1, \textit{node } v_i \textit{ is in transmitting phase in timeslot } t \\ 0, \textit{others} \end{cases} \tag{1}$$

$$r_{it} = \begin{cases} 1, \textit{node } v_i \textit{ is in receiving phase in timeslot } t \\ 0, \textit{others} \end{cases} \tag{2}$$

$u_{ij}(t)$ indicates whether the link $e(i, j)$ between v_i and v_j is active in time slot t, which is defined as follows:

$$u_{ij}(t) = \begin{cases} 1, \textit{the link } e(i, j) \textit{ is busy} \\ 0, \textit{the link } e(i, j) \textit{ is free} \end{cases} \tag{3}$$

To guarantee that the receiving node exists when a node is sending data packet, there must be $u_{ij} = 1$ when $s_{it} + r_{jt} = 2$ or $r_{it} + s_{jt} = 2$, thus

$$u_{ij}(t) = \begin{cases} 1, s_{it} + r_{jt} = 2 \textit{ or } r_{it} + s_{jt} = 2 \\ 0, \textit{others} \end{cases} \tag{4}$$

To match the nodes' forwarding order in the process of routing, the nodes in the network are hierarchical processed. The nodes visited by the traffic flows are layered hop-by-hop in turn. They are layered according to the transmission route of traffic flows from each subnet gateway to the data center. Starting from the data center node, the nodes that have the same hop counts to reach the data center included in the same layer. As shown in **Fig. 3**, v_a, v_b, v_c are the subnet gatewaysand are the sinking nodes of each subnet, D is the data center node. The routing from v_a to D has 2 hops, that from v_b to D has 3 hops, and that from v_c to D has 4 hops, so the nodes are divided into different layers. Nodes v_a, v_b, v_c are on layers 2, 3 and 4, respectively.

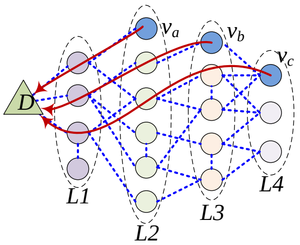

Fig. 3. Hierarchical processed of nodes

According to the above regulations, $neb(i)$ denotes the neighbor node set of the next layer for node v_i: if node v_i belongs to layer L_j, then $neb(i)$ denotes the set of all nodes belonging to layer L_{j+1}.Thus, a valid state vector is defined as

$$c_{it} = \begin{cases} 1, \exists j \in neb(i), makes\ u_{ij} = 1 \\ 0, others \end{cases} \tag{5}$$

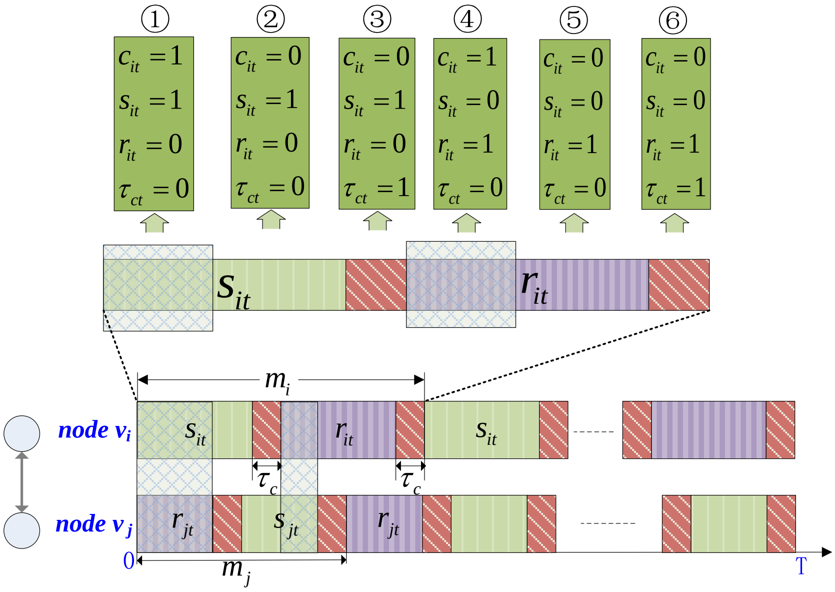

Fig. 4. Constitution of node phases cycle

The related parameters settings as shown in **Fig. 4**. Where s_{it}, r_{it}, c_{it} and τ_{it} are assigned different values in different state. When node v_i is in transmitting phase, $s_{it}=1$, $r_{it}=0$ and $\tau_{it}=0$. If v_i is busy now, $c_{it}=1$, as shown in part ①. If v_i is free, $c_{it}=0$, as shown in part ②. When node v_i is in the switching phase from transmitting to receiving, $s_{it}=1$, $r_{it}=0$, $c_{it}=0$ and $\tau_{it}=1$, as shown in part ③. When node v_i is in receiving phase, $s_{it}=0$, $r_{it}=1$

and $\tau_{it}=0$. If v_i is busy now, $c_{it}=1$, as shown in part ④. In addition, if v_i is free, $c_{it}=0$, as shown in part ⑤. When node v_i is in the switching phase from receiving to transmitting, $s_{it}=0$, $r_{it}=1$, $c_{it}=0$ and $\tau_{it}=1$, as shown in part ⑥.

Let $F=\{f_1, f_2,\cdots, f_m\}$ denotes the traffic flow set and τ_i denotes the cycle of traffic flow $f_i \in F$; N is the number of nodes in set V. We suppose that each node phase cycle length is 2^k $(k=1,2,\cdots)$. Therefore, the node phase cycle length of each node has multiple relationships, and $T=\max\{2^k\}$. For some networks such as WirelessHART and WIA-PA, the cycle length of traffic flow always has this relationship. We call the cycle T in the network super cycle. Because there may be more than one neighbor node for a node, the phase cannot be used with this neighbor node but can be used with another, as shown in **Fig. 5**. It shows the node phase of v_i and its two neighbor nodes v_j and v_k. The blue and red boxes indicate that the node phase of v_i is used with two different neighbor nodes. We can see that the node phase of v_i is used in different time slots with different nodes.

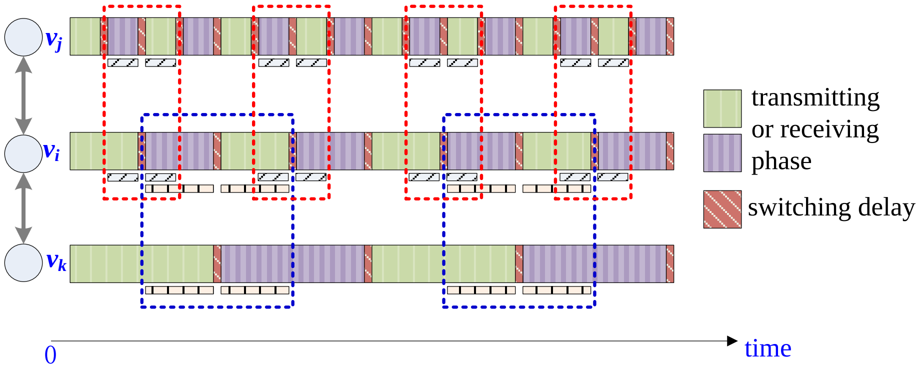

Fig. 5. Node phase utilization with multiple neighbors

Definition 4. Node phase utilization. The node phase utilization is the sum of time slots occupied in each super cycle divided by the total length of the super cycle. Because each node phase cycle length is 2^k $(k=1,2,\cdots)$, the duration of a maximum node phase cycle length $T=\max\{2^k\}$ can cover all situations of each node phase cycle.

If a super cycle has T slots and the cycle length of v_i is T_i, then node v_i contains T/T_i phase cycles. The phase utilization η_i of v_i can be expressed as:

$$\eta_i=\frac{\sum_{t=1}^{T} c_{it}\cdot\left(s_{it}+r_{it}\right)}{T} \tag{6}$$

The phase utilization of the entire network is

$$\eta = \frac{1}{N}\sum_{i=1}^{N}\eta_i = \frac{1}{T\cdot N}\sum_{i=1}^{N}\sum_{t=1}^{T} c_{it}\cdot\left(s_{it}+r_{it}\right) \tag{7}$$

Our problem is as follows. The objective is

$$\max \eta \tag{8}$$

- The Throughput constraint of each node

$$\sum_{t\in T}\sum_{e(i,j)\in i\_in}\left(l_{i\_in}^{t}\times u_{ij}(t)\right)\le\sum_{t\in T}\sum_{e(i,j)\in i\_out}\left(l_{i\_out}^{t}\times u_{ij}(t)\right) \tag{9}$$

It ensures the total traffics that a node received in a super cycle can be transmitted. l_{i_in} denotes the traffic load from other nodes to v_i, and l_{i_out} denotes the traffic load from v_i to other nodes.

- The phase cycle length constraint

$$\forall k\in\Gamma,\ x_k > 2\tau_c \tag{10}$$

The phase cycle length of each node is greater than the switching delay.

- The end-to-end delay constraint

The end-to-end delay of traffics is influenced by the per-hop delay of each node. Therefore, the end-to-end delay constraint is described as follows.

$$\forall f_n \in F, \sum_{i\in\Lambda_{f_n}, i\neq D} d_{i\_to\_neb_i^n} \le \mu_{f_n} \tag{11}$$

Where f_n denotes the traffic flow in flow set F and Λ_{f_n} denotes the node set that traffic flow f_n passes. $d_{i\_to\_neb_i^n}$ is the worst-case delay from node v_i to its next-hop neighbor node in the transmitting route of f_n and can be determined according to the theorem 1. μ_{f_n} is the end-to-end delay demand of traffic flow f_n.

Theorem 1. The worst case delay from the node v_i to the node v_j is $\max\{T_i,T_j\}-\tau$.

Proof. Assume that the phase cycle length of node v_i and node v_j is 2^i and 2^j respectively, and $2^i < 2^j$ initially. The following two cases are discussed.

(1) As shown in **Fig. 6**, if nodes v_i and v_j are neighbor nodes and in different phases initially, the minimum delay occurs when v_i has data to send to v_j and v_j is in receiving phase. The minimum delay is $\min\{T_i,T_j\}-\tau$. When v_i has data to send to v_j and v_j is in the transmitting phase, the delay is maximized as $\max\{T_i,T_j\}-\tau$. For any transmitting order of v_i and v_j, the delay scope is from $\min\{T_i,T_j\}-\tau$ to $\max\{T_i,T_j\}-\tau$.

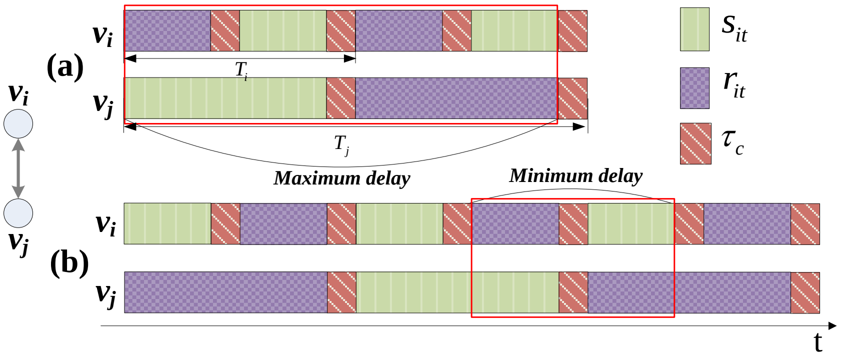

Fig. 6. Delay in different initial phases

(2) If nodes v_i and v_j are neighbor nodes and in the same phase initially, the minimum delay occurs when v_i has data to send to v_j and v_j is in receiving phase. The minimum delay is $\min\{T_i, T_j\} - \tau$. When node v_j has data to send to v_i and v_j is in the transmitting phase, it has the maximum delay $\max\{T_i, T_j\} - \tau$, as shown in Fig. 7. For any transmitting order of v_i and v_j, the delay scope is from $\min\{T_i, T_j\} - \tau$ to $\max\{T_i, T_j\} - \tau$.

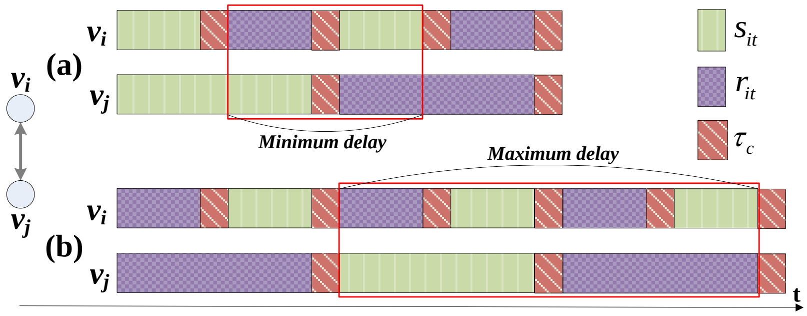

Fig. 7. Delay in same initial phases

4. Problem Solution

In this section, we first reduce the solution space. And then, to obtain the optimum solution of our problem, we propose the algorithm PCL-BAB. Finally, to further reduce the computational complexity, a heuristic algorithm PCL-H is proposed to solve our problem.

4.1 Reducing the Solution Space

We give some theorems to simply the problem. They are as follows.

Theorem 2. In the WiLD network, when the phase cycle length of adjacent nodes is not the same, the node phase utilization depends only on the phase cycle length of the node with minimum node phase cycle. The phase utilization of the node will not change by increasing the phase cycle of the other nodes.

Proof. To simplify the proof, we use two nodes for the discuss and assume that the situation of multiple nodes is the same. The phase cycle length of the two adjacent nodes v_i and v_j is denoted by T_i and T_j respectively. When the phase cycle length of two nodes is unequal, we as well assume that $T_i < T_j$ and $T_j = 2^k \cdot T_i \left(k = 1, 2, \cdots\right)$, as shown in **Fig. 8**. There is $\frac{T_j}{T_i} = 2^k \left(k = 1, 2, \cdots\right)$. In that case, in the phase cycle, half of the phase cycle of node v_i is the same as node v_j, and the other half of the phase cycle of v_i is different with v_j regardless of the initial phase. In that case, the phase utilization is $\frac{\left(\frac{T_j}{2} - \tau\right) \times \frac{T_j}{T_i}}{T_j}$. The general situation is $\frac{\left(\frac{\min\{T_i, T_j\}}{2} - \tau\right) \times \frac{\max\{T_i, T_j\}}{\min\{T_i, T_j\}}}{\max\{T_i, T_j\}} = \frac{\frac{\min\{T_i, T_j\}}{2} - \tau}{\min\{T_i, T_j\}} = \frac{1}{2} - \frac{\tau}{\min\{T_i, T_j\}}$. Obviously, when the phase cycle of the adjacent nodes is not equal, the node phase utilization is determined by the smallest node cycle, and changing the phase cycle of the other nodes has no effect on the phase utilization.

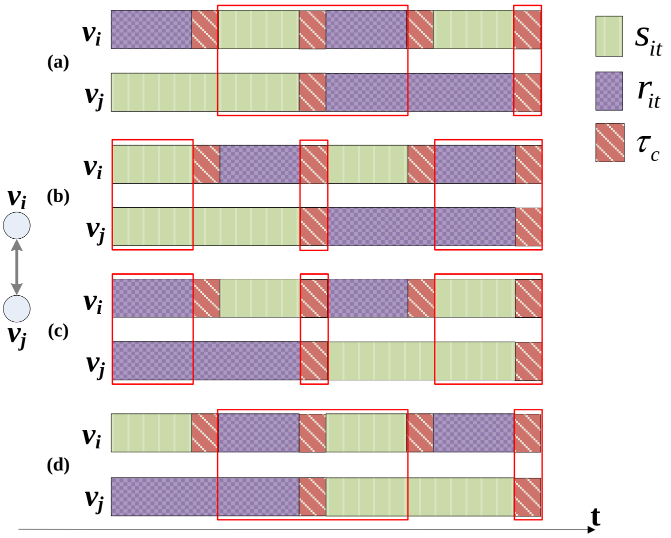

Fig. 8. Node phase utilization in different situations

According to the above analysis and the theorems, the node phase utilization can be represented as

$$\eta_i = \frac{\sum_{t=0}^{T_i^{\max}} \left(s_{it} \cdot r_{jt} + r_{it} \cdot s_{jt}\right) \cdot \left(1 - \tau_{it}\right) \cdot \left(1 - \tau_{jt}\right)}{T_i^{\max}} \tag{12}$$

Where $j \in neb(i)$ and $T_i^{\max} = \max\left\{T_i, T_{neb(i)}\right\}$ is the maximum phase cycle length in v_i and $neb(i)$.

Theorem 3. Keep the phase cycle length of the other nodes constant and increase the phase cycle length of one node. When the node phase cycle changing from unequal to equal with other nodes, it will increase the node phase utilization; otherwise, it will reduce the node phase utilization.

Proof. To simplify the proof, we still use two nodes for our discussion.

First, when the phase cycle length of two nodes is equal, that is $T_i = T_j$, we discuss the problem in two cases. If the initial phases of node v_i and v_j are the same, as shown in **Fig. 9(a)**, then the node phase utilization is 0. At this time, the nodes are unable to communicate with each other, so we don't consider this situation in an actual network. Conversely, if the initial phases are opposite, as shown in **Fig. 9(b)**, the node phase utilization is $\frac{T_i-2\tau}{T_i}=1-\frac{2\tau}{T_i}$.

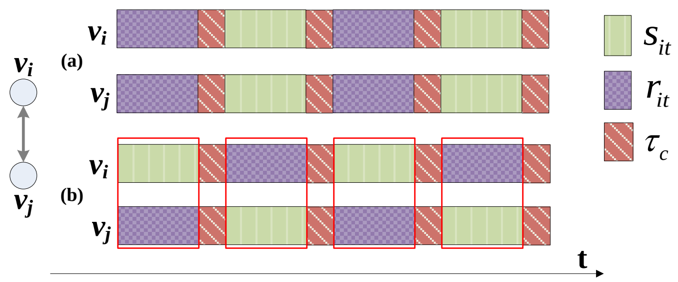

Fig. 9. Delay in different initial phases with same cycle length

Second, When the phase cycle length of two nodes is unequal, we also assume that $T_i < T_j$. From the previous theorem 1, we known that the node phase utilization is $\frac{1}{2}-\frac{\tau}{\min\{T_i,T_j\}}$.

Third, we should consider the changes of the node phase utilization when the phase cycle transforming between equal and unequal.

We first consider the condition that the phase cycle lengths of two nodes are unequal at start time, but are equal after the phase cycle length of one node increased. Assume that $T_i = 2^k \cdot T_j \left(k=1,2,\cdots\right)$ at the beginning. Thus, $T_i = T_j^{'}$ after the phase cycle length of node v_j increased. At this time,

$$\left(\frac{1}{2}-\frac{\tau}{\min\left\{T_i,T_j\right\}}\right)-\left(1-\frac{2\tau}{T_i}\right)=\frac{1}{2}-\frac{\tau}{T_j}-1+\frac{2\tau}{T_i}=\left(\frac{2}{T_i}-\frac{1}{T_j}\right)\cdot\tau-\frac{1}{2}=\frac{\tau}{T_j}\cdot\left(\frac{1}{2^{k-1}}-1\right)-\frac{1}{2}.$$

Because $\frac{1}{2^{k-1}}-1\le 0$, it is obvious that the upper formula <0. Thus, the node phase utilization will increase in this situation.

We then consider the condition in which the phase cycle lengths of two nodes are equal at start time but are unequal after the phase cycle length of one node increased. Assume that $T_i = T_j$ at the beginning. Thus, $T_j^{'} = 2^k \cdot T_i \left(k=1,2,\cdots\right)$ after the phase cycle length of node v_j increased. Because $T_i > 2\tau$, $\left(1-\frac{2\tau}{T_i}\right)-\left(\frac{1}{2}-\frac{\tau}{\min\left\{T_i,T_j^{'}\right\}}\right)=1-\frac{2\tau}{T_i}-\frac{1}{2}+\frac{\tau}{T_i}=\frac{1}{2}-\frac{\tau}{T_i}>0$. So $\left(1-\frac{2\tau}{T_i}\right)-\left(\frac{1}{2}-\frac{\tau}{\min\left\{T_i,T_j^{'}\right\}}\right)>0$; that is, the node phase utilization will decrease in this situation.

According to the theorem 2, when the node phase cycle length of v_i and v_j are $T_i = 2^a$, $T_j = 2^b \left(a < b\right)$, respectively, the node phase utilization is $\eta = \mu$. Then, when $T_i = 2^k, k = a+1, a+2, \cdots, b-1$, the node phase utilization remains the same. According to theorem 3, when T_i increases to $T_i' = T_j = 2^b$, the value of η will increase and change the node phase utilization. Based on these theorems, in the following method, we can omit some unnecessary calculations of the node phase utilization, which will reduce the number of calculations in the algorithm. Thus, in the process of the branching, when the phase cycle length of the same node is changing from 2^a to 2^b $(a < b)$ and is not equal to its neighbor nodes, if the phase cycle length of its neighbor nodes are not change, we need only calculate the node phase utilization at point 2^a and 2^b, and the values between 2^a to 2^b are skipped.

4.2 A Method Based on Branch and Bound

A branch and bound algorithm is a method to solve the optimization problems that is based on three key factors: branching, bounding and elimination.

Branching is the process of cutting the feasible region of the length of switching time into smaller parts. Bounding is the process of determining the bounds of the objective function in these parts. Elimination is the process of dropping the parts that exceed the bounds. When the algorithm runs to a certain stage, the parts that meet the elimination rules will be dropped. Because these parts do not contain the optimal solution, they do not require further branching. The elimination rule in this paper is that the actual end-to-end delay of the traffic exceeds the delay demand.

We chose the branch and bound algorithm because it can obtain an optimal solution. The pseudo-code of the algorithm is shown in **Algorithm 1**, where N is the number of nodes in the network, m is a threshold of k in node phase cycle length 2^k, F is the set of traffic flows, and $\mu_{f_i} \in \mu_F$ is the end-to-end delay demand of the traffic flow $f_i \in F$. d_F is the actual end-to-end delay of traffic flow f_i. η_τ is the user setting threshold value of η.

The main idea of the algorithm is as follows.

- Preparation.

Users set an acceptable phase cycle length utilization threshold η_τ and the network is initialized (**Algorithm 1**, line 1). The routings from each subnet gateway node to the sink node in data center are determined according to a classical routing algorithm, such as dijkstra or floyd. The phase cycle length of all nodes is set to 2^1 initially.

Algorithm 1 The Branch and Bound Solving Method

```
Require: $N$, $f_i \in F$, $\mu_{f_i} \in \mu_F$, $d_F$, $\eta_\tau$
Ensure: $\eta$
1: network initialization;
2: BB(level, $\Gamma$);
return $\eta \leftarrow \max\{\eta\}, \Gamma$
```

We obtain the max-cut of the network according to the biggest cut algorithm Algo-2 [18]. The initial phase of the nodes in the cut set with data center node is set to receiving, and the initial phase of all nodes is another cut set is set to transmitting. By using Algo-2, if the network topology can form a binary graph, it can divide the nodes of the network into two binary subsets.

- Branching.

Assume that the sinking node in data center is v_1. Starting from v_1 to v_N, list all possible phase cycle length allocations of each node (from 2^1 to 2^m). The same node (such as v_i) with different phase cycle length allocation is divided into different branches. Then, the next node (v_{i+1}) with a different phase cycle length allocation is set as the leaf of each branches, as shown in **Fig. 10**.

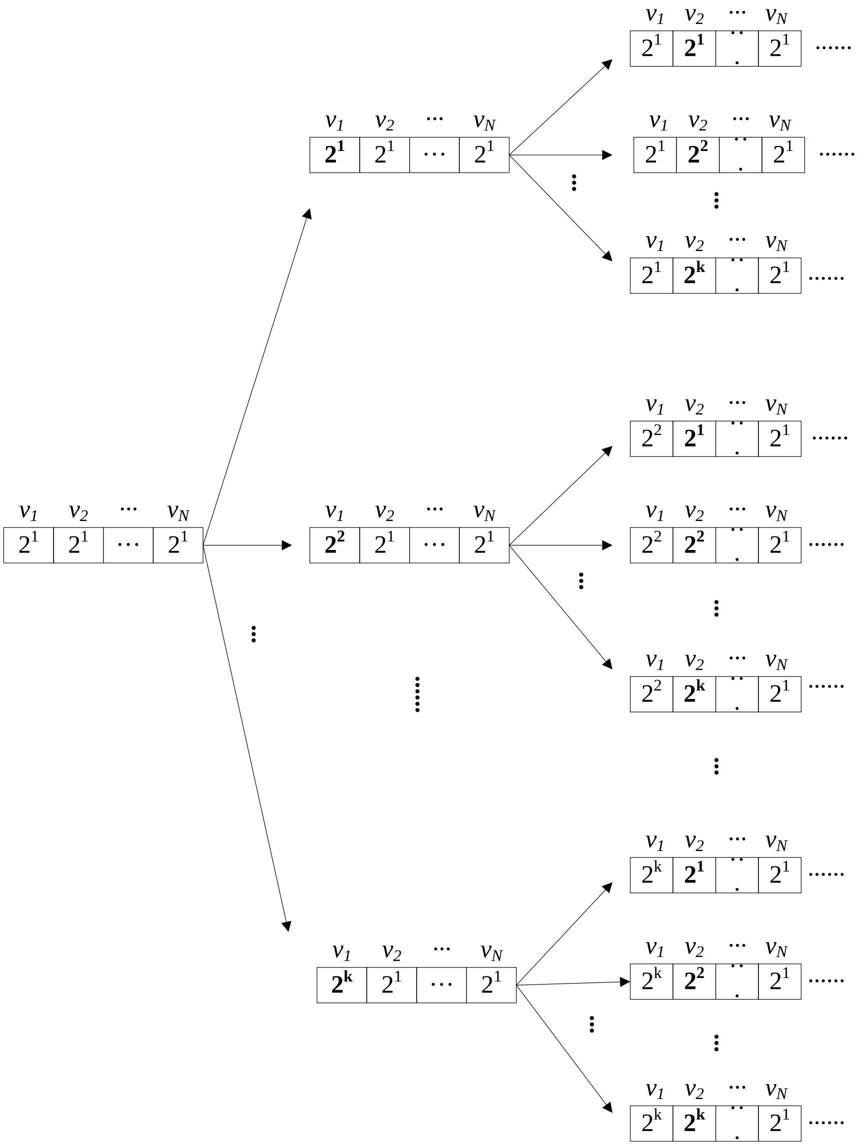

Fig. 10. An example of branching

The branch and bound operation is performed according to a recursive function *BB*(*level*□Γ) (as shown in **Algorithm 2**), where *level* from 0 to *N* denotes the branch level and $\Gamma = \{T_1, T_2, \cdots, T_N\}$ is the cycle length set of each node. When *level*=0, $\Gamma = \{2^1, 2^1, \cdots, 2^1\}$. According to theorems 2 and 3 and the analysis above, we can skip the calculation of some η (**Algorithm 2**, lines 1-2). In this way, we can save some computing time.

```
Algorithm 2 function BB (level, $\Gamma$)
1: if $\eta$ need calculate then
2:   calculate $\eta$
3:     if $\eta < \eta_\tau$ then
4:       record $\eta$;
5:     end if
6: end if
7:for $i = 1$ to $N$ do
8:  if $\forall d_{f_i} \in d_F, d_{f_i} \le \mu_{f_i}$ then
9:    for $j = 1$ to $m$ do
10:     $T_i = j$;
11:     BB($i$, $\Gamma$);
12:   end for
13:  end if
14:end for
return $\eta$
```

● Bounding.

For each case listed, we should analyze whether it meets the elimination rules based on the branch and bound algorithm. When it meets the rules, this branch is eliminated. When it does not meet the rules, we return to branching and continue the operations (**Algorithm 2**, lines 7-14). In the process of branching, if the phase cycle length η is less than the user's set threshold η_τ, we record this η and continue the branching operations; otherwise, we continue the branching operations without recording this η (**Algorithm 2**, lines 3-5).

The number of iterations of **for** loop in line 7 and line 9 is $O(|N|)$ and $O(|m|)$, respectively. And it is a recursive method and call itself in line 11. So the time complexity of this algorithm is $O(|m^N|)$.

4.3 A Heuristic Method

To reduce both the complexity of the algorithm and the search space, a heuristic search algorithm PCL-H is designed in this paper. We still assume that the maximum acceptable phase cycle length of the network is $2^m = \max\left\{2^k\right\}$.

The pseudo-code of the algorithm is shown in **Algorithm 3**. The basic idea of the algorithm is as follows.

- Preparation.

The preparation stage of PCL-H is the same as PCL-BAB, except in this algorithm, the phase cycle length of all nodes is initially set to 2^m (**Algorithm 3**, line 1).

- Link time allocation.

Calculate the end-to-end delay of all traffics in the network according to the transmission route of each traffic flow(**Algorithm 3**, line 2). If there is a traffic whose transmission delay through the network is greater than the end-to-end delay demand, wefind the link with the most traffic flows across in this traffic's transmission route and divide the phase cycle length of the two nodes connected by this link by two. When multiple links have the same number of traffic flows, choose the link with the traffic that has the least time remaining until the deadline (**Algorithm 3**, lines 3-7).

- Calculate the end-to-end delay of each traffic.

According to theorem 3, when the phase cycle length of two nodes is equal, it has the maximum phase utilization. Therefore, if the end-to-end delay of each traffic can be satisfied, find the node through which only one traffic flow passes, and let the phase cycle length of this node equal the phase cycle length of an adjacent node that has other traffics passing. Otherwise, find the traffic flows that cannot meet the end-to-end delay and put them in a new traffic set. Set these traffic flows as the network traffic and ignore the traffic flows whose real time demands are satisfied. Then, conduct the link time allocation again until the end-to-end delay of each traffic can be satisfied (**Algorithm 3**, lines 8-11).

- Calculate the node phase utilization.

Calculate the node phase utilization (**Algorithm 3**, line 12), and the algorithm terminates.

Algorithm 3 The Heuristic Search Method

```
Require: $N$ , $f_i \in F$ , $\mu_{f_i} \in \mu_F$ , $d_F$ , $\eta_\tau$
Ensure: $\eta, \Gamma$
1: network initialization;
2: $\forall d_{f_i} \in d_F$ , calculate $d_{f_i}$ ;
3: while $F \neq \phi$ do
4: find the link $e(i,j)$ with maximum number of traffic flows passes (denoted by $e_{ij}^{\max\_F}$ );
5:   if the number of $e_{ij}^{\max\_F} > 1$ then
6:     choose the link with least remaining deadline traffics pass through;
7:   end if
8: $\max\left\{T_i, T_j\right\} \leftarrow \max\left\{T_i, T_j\right\}/2$ ;
```

```
9: $\forall d_{f_i} \in d_F$, calculate $d_{f_i}$;
10: $F \leftarrow F - \{d_{f_i} \le \mu_{f_i}\}$;
11:end while
12: calculate $\eta$;
return $\eta, \Gamma$
```

The number of iterations of **while** loop in line 3 is $O(|F|)$ and in line 4 is $O(|E|)$, So the time complexity of **Algorithm 3** is $O(|F| \cdot |E|)$.

Then, for the computation time depends on many things such as computer processor performance, optimization of codes and so on. Therefore, the results for computation time of different algorithms are not the best indicator. But, as far as I know, there is no solving method with the same objective and constraints as this paper that based on the real time demands of traffics to minimize the switching overhead. Thus, we just compare the number of parameters that each algorithm considered. We list the comparison of proposed algorithms PCL-BAB and PCL-H with traditional algorithms Algo-2, Algo-d in **Table 1**. From the table we see that, the time complexity of alogrithms PCL-H, Algo-2 and Algo-d are all relatively low. The time complexity of PCL-BAB is higher, but with a optimal solution. The algorithms PCL-BAB and PCL-H consider more parameters than other two algorithms.

Table 1. Time Complexity Compare

Algorithm	Time complexity	End-to-end delay	Throughput	Switching overhead	Real-time
PCL-BAB	$O(\|m^N\|)$	√	√	√	√
PCL-H	$O(\|F\| \cdot \|E\|)$	√	√	√	√
Algo-2	$O(\|V\|^2)$	√	√	×	×
Algo-d	$O(\|E\|)$	√	√	×	×

5. Simulation and Results

We want to satisfy the real time demand of the traffic while minimize the phase switching overhead, and just need the traffic with strict end-to-end delay demand. Because the CBR (Constant Bit Rate) [20] flows with strict end-to-end delay demands can simulate the real industrial traffics, we use simulation instead of experiment. Besides, with simulation, we can verify more situations which can better explain the universality of our algorithms.

The simulation software Matlab with the Matgraph toolbox was used to implement the algorithm. With the CBR, we can send data at a constant rate and size. By changing both the network size and the number of traffic flows, we count the node phase utilization and the average cycle length of the network and the changes in search time during different CBR flows and numbers of WiLD network nodes, respectively. The average phase cycle length is the average length of the phase cycle of all nodes in the network.

The number of CBR flows changes from 1 to 10, and the number of WiLD network nodes changes from 4 to 20. Finally, the algorithm proposed in this paper was compared with the traditional traversing search method. The results are shown in **Figs. 6-13**.

5.1 The Results in Different Number of Traffic Flows

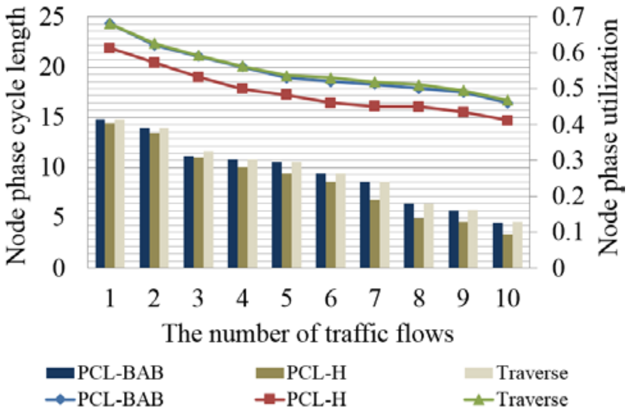

Fig. 11. Changes in the average node phases cycle length with the number of traffic flows (the number of nodes is 10)

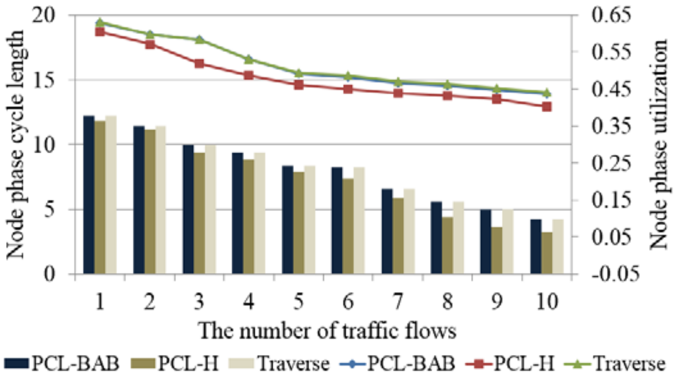

Fig. 12. Changes in the average node phases cycle lengths with the number of traffic flows (the number of nodes is 20)

Fig. 11 and **Fig. 12** show the changes in the average node phase cycle length in the network with different number of traffic flows when the number of nodes is 10 and 20, respectively. The bar charts represent the node phase cycle lengths, and the line charts represent the node phase utilization. From these two figures, we can see that with an increase in the number of traffic flows, the average phase cycle length and the node phase utilization with algorithms PCL-BAB, PCL-H and traverse all gradually decrease. Because of the increase in the number of traffic flows, each node has more traffics passing. There may be different delay demands for different traffic flows in one node, and the phase cycle length of this node must therefore meet the minimum delay demand of the traffics. With the increase in traffic flows, the minimum delay demand of the traffics in a node may decrease, and the node phase cycle length must be decreased frequently to meet the real time demands of different traffic flows. However, the node phase cycle length and the node phase utilization with algorithms PCL-H is near the optimal solution obtained by PCL-BAB and traversal search.

5.2 The Computing Time with Different Number of Traffic Flows

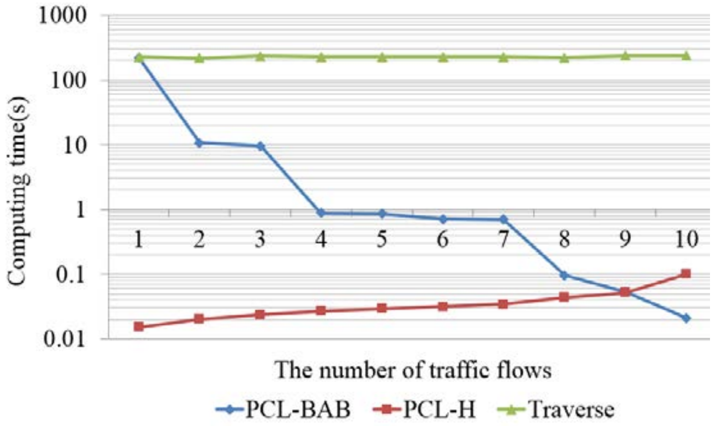

Fig. 13. Changes in computing time with the number of traffic flows (the number of nodes is 10)

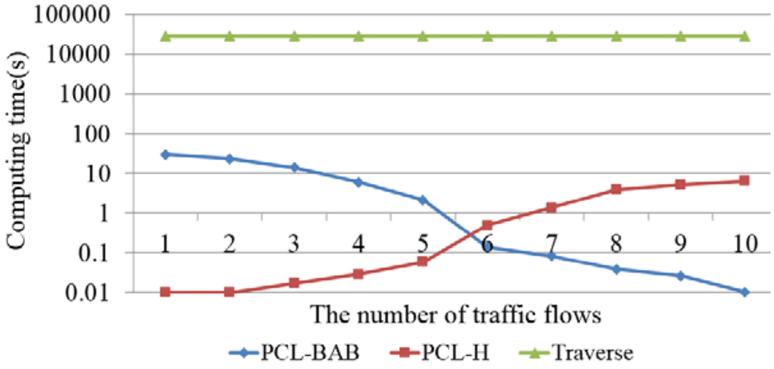

Fig. 14. Changes in computing time with the number of traffic flows (the number of nodes is 20)

Fig. 13 and **Fig. 14** show the changes in computing time with different numbers of traffic flows when the number of nodes is 10 and 20, respectively. From the results, we can see that the computing time of algorithm PCL-BAB decreases rapidly, the computing time with algorithm PCL-H increases gradually, and the traversal search is almost unchanged. With an increase in the number of traffic flows, more nodes are constrained by the real-time traffic demands. Therefore, the cycle length of these nodes should decrease to meet the delay demands. There are more branches that cannot meet the end-to-end delay demands of the traffic flows that were eliminated in algorithm PCL-BAB. Thus, the computing operations are decreased, and the computing time is saved. However, the traversal search will compute all situations, regardless of the number of traffic flows.

5.3 Results for Different Network Scales

Fig. 15 and **Fig. 16** show the changes in the average node phase cycle length of the network with different network scales when the number of traffic flows is 3 and 8, respectively. From these results, we can see that the overall trend in the average phase cycle length and the node phase utilization with algorithms PCL-BAB, PCL-H and traverse are decreasing gradually. However, in Fig. 16, the average node phase cycle length and node phase utilization first increase and then decrease. When the number of traffic flows is higher while the network is small, all the nodes may be constrained by the real-time traffic demands, and most nodes have more than one traffic flow passing. Thus, when the number of nodes increases, the node phase cycle length will increase in a certain range. However, with an increase in the number of nodes, there are more nodes that are not constrained by the real-time traffic demands. Thus, the average node phase cycle length of the network increases. When the number of nodes increases to a certain value, the hop is the main factor that affects the end-to-end delay of the traffics. The nodes need to reduce the node phase cycle length so they can change their phase frequently and thereby meet the real-time demands of different traffic flows. Thus, the node phase cycle length will decrease gradually.

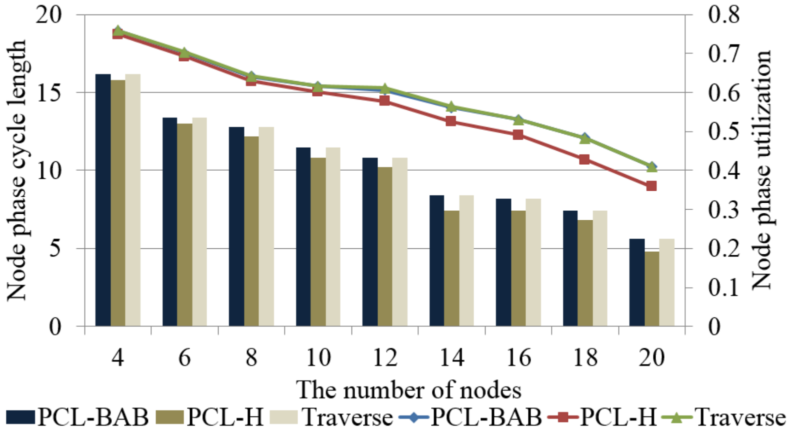

Fig. 15. Changes in the average node phases cycle length with network scale (the number of traffic flows is 3)

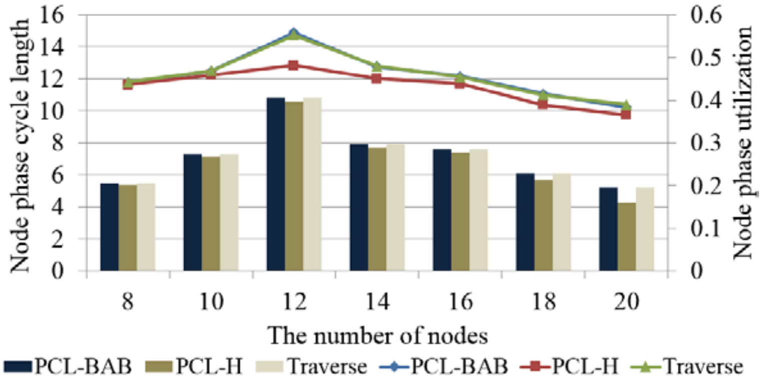

Fig. 16. Changes in the average node phases cycle length with network scale (the number of traffic flows is 8)

5.4 Computing Time with Different Network Scales

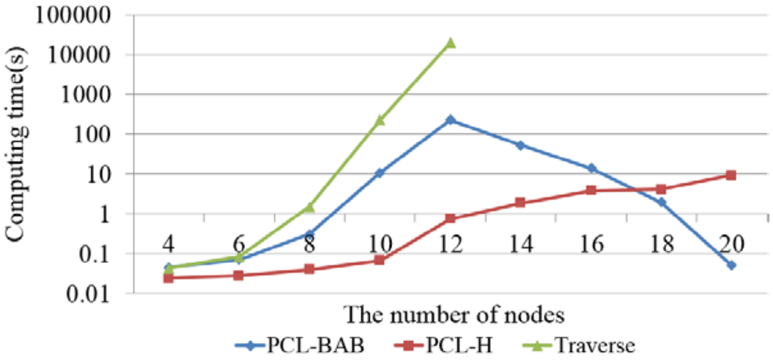

Fig. 17. Changes in computing time with the network scale (the number of traffic flows is 3)

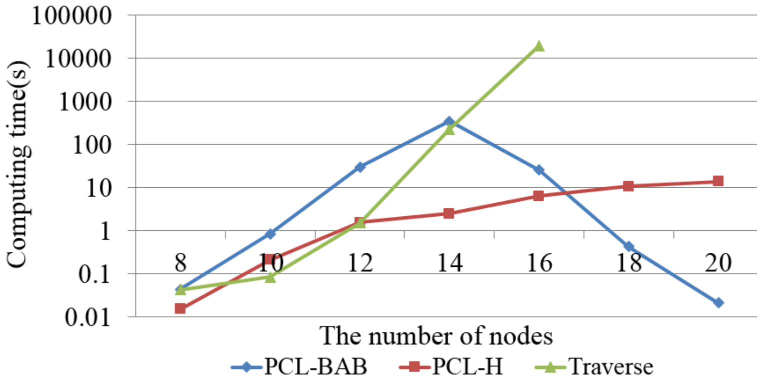

Fig. 18. Changes in computing time with the network scale (the number of traffic flows is 8)

When the number of traffic flows is 3 and 8, the changes in the computing time under different number of nodes are shown in **Fig. 17** and **Fig. 18**, respectively. If the number of nodes is more than 16, the computing time of traversal search is too large (may reach to 10^{10} seconds). So we show only the traversal results under different nodes number from 8 to 16. From the results, we can see that during the initial increase in the number of nodes, the computing time with algorithms PCL-BAB, PCL-H and traverse all increase. The more nodes there are, the more states the algorithm has to search. Then, when the number of nodes reaches a certain stage, the alternative phase cycle length of a node decreases, so the number of branches that can be eliminated increases, and the computing time of algorithm PCL-BAB decreases. The computing time of algorithms PCL-BAB and PCL-H is in an acceptable range.

6. Conclusions

With an industrial control network, the manufacturing process in each operating conditions can be timely grasped, and the operation of each process is optimize. This can ensure the production quality, reduce the production costs, improve the level of operation and management, and stabilize long-term production. It conforms to the development trends found in information integration system and control & management integration.

In this paper, we research the time assigning and switching mechanism based on the classic 2P-MAC protocol in the IEEE 802.11 WiFi-based Long Distance (WiLD) network time-division patterns. The object of our research is to provide a delay guarantee for the high real-time, cyclical traffic flows in industrial applications. The simulation results show that the algorithm PCL-BAB can obtain the optimum solution, and the results of the heuristic algorithm PCL-H are close to the optimal solution while the computing time is reduced.

References

[1] A. Saifullah, X. You, L. Chenyang and C. Yixin. “End-to-end communication delay analysis in industrial wireless networks,” *IEEE Transactions on Computers*, vol. 64, no. 5, pp.1361-1374, 2015. Article (CrossRef Link)

[2] Y. Ming and Z. Xinlong, “Time division task scheduling algorithm for network transmission,” *Information and Control (in Chinese)*, pp.660-666, January 1-6, 2015. Article (CrossRef Link)

[3] M. I. Hussain, S.K. Dutta, N. Ahmed and I. Hussain, “A wifi-based reliable network architecture for rural regions,” *ADBU Journal of Engineering Technology*, vol. 3, no. 1, pp.1-6, 2015. Article (CrossRef Link)

[4] K. Cho, S. Jeon, J. Cho and B. Lee, “ISRMC-MAC: Implementable Single-Radio, Multi-Channel MAC Protocol for WBANs,” *KSII Transactions on Internet and Information Systems*, vol. 10, no. 3, pp. 1052-1070, 2016. Article (CrossRef Link)

[5] D. Zhibin, Z. Zenghua, J. Quan, Z. Lianfang, S. Yantai and Y. Oliver. “Energy-efficient rate adaptation for outdoor long distance wifi links,” in *Proc. of Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference*, pp. 271-276, April 10-15, 2011. Article (CrossRef Link)

[6] S. Nedevschi, R. K. Patra, S. Surana and et al., “An adaptive, high performance mac for long-distance multihop wireless networks,” in *Proc. of the 14th ACM international conference on Mobile computing and networking*, pp. 259-270, September 14-19, 2008. Article (CrossRef Link)

[7] B. Raman and K. Chebrolu. “Design and evaluation of a new mac protocol for long-distance 802.11 mesh networks,” in *Proc. of the 11th annual international conference on Mobile computing and networking*, ACM, pp.156-169, August 28-September 2, 2005. Article (CrossRef Link)

[8] R. K. Patra, S. Nedevschi, S. Surana and et al. “Wildnet: Design and implementation of high performance wifi based long distance networks,” in *Proc. of Networked Systems Design and Implementation (NSDI 2007)*, pp. 1, April 11-13, 2007. Article (CrossRef Link)

[9] S. Surana, R. Patra, S. Nedevschi, and E. Brewer. “Deploying a rural wireless telemedicine system: Experiences in sustainability,” *Computer*, vol.41, no.6, pp.48-56, 2008. Article (CrossRef Link)

[10] Y. Ben-David, M. Vallentin, S. Fowler, and E. Brewer. “Jaldimac: taking the distance further,” in *Proc. of the 4th ACM workshop on networked systems for developing regions (NSDR)*, pp. 2, June 15, 2010. Article (CrossRef Link)

[11] Z. Zenghua, H. Ming, Z. Jie and Z. Lianfang. “Qos routing and traffic scheduling in long-distance 802.11 wireless mesh networks,” *Chinese Journal of Electronics*, vol.21, no.2, pp. 313-317, 2012. Article (CrossRef Link)

[12] I. Hussain, N. Ahmed, D. K. Saikia, and N. Sarma. “A qos-aware multipath routing protocol for wifi-based long distance mesh networks,” in *Proc. of Emerging Technology Trends in Electronics, Communication and Networking (ET2ECN), 2014 IEEE 2nd International Conference*, pp. 1-8. Dec. 26-27, 2014. Article (CrossRef Link)

[13] I. Hussain, D. K. Saikia, N. Sarma, and N. Ahmed, “A fine-tuned packet scheduling for wifi-based long distance networks,” in *Proc. of Applications and Innovations in Mobile Computing (AIMoC)*, 2014, pp. 97-103, Feb. 27-Mar.1, 2014. Article (CrossRef Link)

[14] K. Greene, “Long-Distance Wi-Fi: Intel has found a way to stretch a Wi-Fi signal from one antenna to another located more than 60 miles away,” Article (CrossRef Link)

[15] L. Ming, S. Salinas, L. Pan, S. Jinyuan and H. Xiaoxia, “Mac layer selfish misbehavior in IEEE 802.11 ad hoc networks: Detection and defense,” *IEEE Transactions on Mobile Computing*, vol.14, no.6, pp.1203-1217, 2015. Article (CrossRef Link)

[16] Y. Jing , E. Bae, T. Xue-Cheng, and Y. Boykov, “A spatially continuous max-flow and min-cut framework for binary labeling problems,” *Numerische Mathematik*, vol.126, no.3, pp.559-587, 2014. Article (CrossRef Link)

[17] K. W. Chin, S. Soh, and C. Meng, “A novel scheduler for concurrent tx/rx wireless mesh networks with weighted links,” *Communications Letters, IEEE*, vol.16, no.2, pp.246-248, 2012. Article (CrossRef Link)

[18] H. Y. Loo, S. Soh and K. W. Chin, “On improving capacity and delay in multi tx/rx wireless mesh networks with weighted links,” in *Proc. of 19th Asia-Pacific Conference on Communications (APCC)*, pp.12-17, Aug. 29-31, 2013. Article (CrossRef Link)

[19] H. Wang, K. W. Chin, S. Soh and R. Raad, “A distributed maximal link scheduler for multi tx/rx wireless mesh networks,” *IEEE Transactions on Wireless Communications*, vol.14, no.1, pp.520-531, 2015. Article (CrossRef Link)

[20] X. Deng, L. He, Q. Liu and et al., “EPTR: expected path throughput based routing protocol for wireless mesh network,” *Wireless Networks*, vol.22, no.3, pp.839-854, 2016. Article (CrossRef Link)

Jintao Wang graduated from Heilongjiang University, China, in 2010. He received the M. Sc. Degree from Northeastern University, China, in 2012. He is currently a Ph. D. candidate at Shenyang Institute of Automation, Chinese Academy of Sciences. His research interests include industrial control network and wireless mesh networks.

Xi Jin graduated from Northeastern University, China, in 2006. She received the M. Sc. and Ph. D. degrees from NEU in 2008 and 2013, respectively. She is currently an associate professor at Shenyang Institute of Automation, Chinese Academy of Sciences. Her research interests include wireless sensor networks and real-time systems, especially the realtime scheduling algorithms, and worst case end-toend delay analysis. Corresponding author of this paper.

Peng Zeng received his Ph. D. degree from Shenyang Institute of Automation, Chinese Academy of Sciences. He is currently a professor at Shenyang Institute of Automation, Chinese Academy of Sciences. His research interests include industrial communication and wireless sensor networks.

ZhaoWei Wang received the B.S. degree in measurement & control technology and instrumentation from Zhengzhou University,in 2012. Currently, he is pursuing the Ph.D. degree in control theory and control engineering at Shenyang Institute of Automation (SIA), Chinese Academy of Sciences (CAS). His research interests include wireless sensor networks and industrial wireless networks

Ming Wan received the Ph. D. degrees from Beijing Jiaotong University. He is currently an associate professor at Shenyang Institute of Automation, Chinese Academy of Sciences. His research interests include the future network and information security, the information security of industrial control systems.