AN INTERFERENCE ELIMINATION METHOD FOR DECENTRALIZED SLOT SYNCHRONIZATION IN TDMA-BASED WIRELESS AD HOC NETWORK

*Qi Yang, Jianghong Shi**

Information Science and Technology College of Xiamen University, Xiamen, China

ABSTRACT

Time slot synchronization is very important for a TDMA-based Ad Hoc network. An existing method to achieve time slot synchronization in ad hoc network is the decentralized slot synchronization algorithm. However, interference nodes in the network may lead to un-convergence of this algorithm. In this paper, a method for eliminating the interference in decentralized slot synchronization is presented. To evaluate the performance of the proposed mechanism, the influence to the convergence of slot synchronization in the network is investigated. Simulation results indicate that, with this method, the interference is effectively eliminated from the computation of decentralized slot synchronization.

***Index Terms*—** Ad hoc networks, Slot synchronization, TDMA, Interference

1. INTRODUCTION

Ad hoc network is a distributed multi-hop wireless packet network independent of any infrastructure. For communication system based on slot Time Division Multiple Access (TDMA) scheme, time slot synchronization is required among the nodes in the network [1]. Besides obtaining the slot synchronization reference from the Global Position System (GPS), some decentralized slot synchronization schemes are proposed in [2][3][4]. These schemes share a common feature that slot synchronization is achieved by shifting the local slot reference of each node towards a weighted average of the slot reference of the surrounding nodes.

All the decentralized slot synchronization algorithms are based upon the assumption that there are no interference nodes in the network, and all nodes in the networks can comply with the algorithm. This may not be reasonable assumption since interference nodes and mal-functional nodes un-complying with the algorithm in the network is inevitable, which may degrade the accuracy of the algorithm, and worse still, may lead to un-convergence of the algorithm. Only few approaches for eliminating the interference nodes from decentralized slot synchronization algorithm have been published so far. A weighted parameter selection scheme to deal with malfunctioning nodes is proposed in [5], whereas it only decreases the infection of the interference node. In this paper, a method for eliminating the interference from decentralized slot synchronization in TDMA-based ad hoc network is proposed.

The rest of this paper is organized as follows. In Section 2, the principle of decentralized slot synchronization algorithm for wireless ad hoc network is discussed. In Section 3, the factors that infecting the convergence are discussed and the interference elimination method is also derived. In Section 4, the simulation results of proposed method for different scenarios are given. In Section 5, the conclusion is presented.

2. DECENTTALIZE SLOT SYNCHRONIZATION

In decentralized slot synchronization algorithm [3], each node in the network periodically computes out the time slot differences between its local time slot reference and those of its neighboring nodes, then the local time slot adjustment value is calculated based on the weighed average of these slot differences. Each node in the network adjusts its local time slot reference by the slot adjustment value computed, and their local time slot reference will shift towards a common constant value. Thus the global time slot synchronization is archived.

For example, Figure1 shows that node 5 computes out the time slot differences between its local slot reference and those of its neighboring nodes (i.e. node 3, node 6 and node 9, which is $\Delta t_{53}^{(n)}$, $\Delta t_{56}^{(n)}$ and $\Delta t_{59}^{(n)}$ respectively).

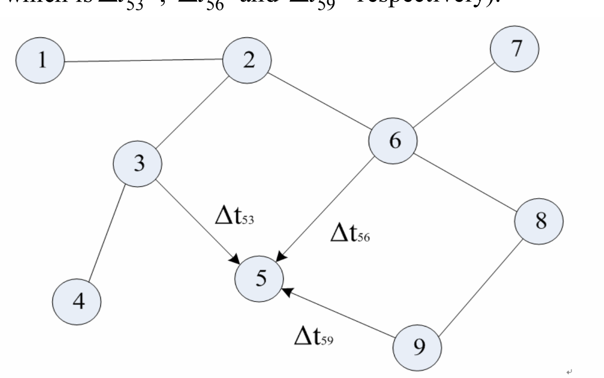

Figure 1 Synchronization Process of a Node

The slot adjustment value of node 5 is calculated:

$\varepsilon_5^{(n)} = w_{55}^{(n)} \cdot 0 + w_{53}^{(n)} \Delta t_{53}^{(n)} + w_{56}^{(n)} \Delta t_{56}^{(n)} + w_{59}^{(n)} \Delta t_{59}^{(n)}$

Where: $w_{55}^{(n)} + w_{53}^{(n)} + w_{56}^{(n)} + w_{59}^{(n)} = 1$

Node 5 adjusts its local time slot reference according to the adjustment computed above: $t_5^{(n+1)} = t_5^{(n)} - \varepsilon_5^{(n)}$.

For the convenience of theory analysis, we consider that the adjustment value $\varepsilon_i^{(n)}$ can be modeled as the weighted average of differences between local slot reference of node i and those of all other nodes in the network, where the weight of non-connective node is zero, which is written as:

$$\varepsilon_i^{(n)} = \sum_{k=1}^{N} w_{ik}^{(n)} \Delta t_{ik}^{(n)}, (k = 1,2,...,N) \quad (1)$$

where: $\sum_{k=1}^{N} w_{ik}^{(n)} = 1$

The weighted average $\varepsilon_i^{(n)}$ is the adjustment to modify the local time slot reference of node i :

$$t_i^{(n+1)} = t_i^{(n)} - \varepsilon_i^{(n)} \quad (2)$$

The slot reference of each node after the n^{th} adjustment is:

$$\begin{aligned} t_i^{(n+1)} &= t_i^{(n)} - \varepsilon_i^{(n)} = t_i^{(n)} - \sum_{j=1}^{N} w_{ij}^{(n)} \Delta t_{ij}^{(n)} \\ &= t_i^{(n)} - \sum_{j=1}^{N} w_{ij}^{(n)} (t_i^{(n)} - t_j^{(n)}) = \sum_{j=1}^{N} w_{ij}^{(n)} t_j^{(n)} \end{aligned} \quad (3)$$

This process is repeated periodically for each node. Suppose that, all the nodes in the network comply with the algorithm $\varepsilon_i^{(n)}$ will approach 0 after numerous times of iteration. In other words, local time slot reference of each node will converge to a common constant value. The convergence of the algorithm is illustrated in [2].

3. INTERFERENCE ELIMINATION METHOD

In practical networks, nodes that un-complying with the decentralized slot synchronization algorithm are inevitable, which may lead to un-convergence of the algorithm. These nodes can be deemed to interference nodes. There are three classes of interference nodes, that is stationary nodes, new coming nodes and vibrating nodes. To guarantee the robustness of the algorithm, the interference nodes in the network must be eliminated from the calculation in each iteration computing.

Firstly, if there are more than two stationary nodes whose time slot reference doesn't change during each adjustment, the algorithm may not converge.

In a network with N nodes, assume that node p and node q can't adjust its local time reference, which is t_p and t_q respectively. The slot reference of any arbitrary node in the network at the next iteration computation is:

$$t_i^{(n+1)} = \sum_{j=1}^{N} w_{ij}^{(n)} t_j^{(n)} = \sum_{j=1, j\neq p, j\neq q}^{N} w_{ij}^{(n)} t_j^{(n)} + w_{i,p} t_p + w_{i,q} t_q$$

Suppose that the time reference of the rest nodes in the network is converge to a value (suppose it is t), That is:

$t_j^{(n)} = t$, $(j = 1,2 \cdots N, j \neq p, j \neq q)$

Hence: $t_i^{(n+1)} - t_i^{(n)} = w_{i,p}(t_p - t_i^{(n)}) + w_{i,q}(t_q - t_i^{(n)})$

Namely, $\varepsilon_i^{(n)} = w_{ip}^{(n)} \Delta t_{ip}^{(n)} + w_{iq}^{(n)} \Delta t_{iq}^{(n)}$

Apparently, the necessary condition for slot references of all other nodes in the network converging to a constant value is:

$w_{ip} \Delta t_{ip} + w_{iq} \Delta t_{iq} = 0$

It's well see that we must find out the node that can't adjust its local slot reference in the network, and eliminate it from the computation. If $t_k^{(n)} = t_k^{(n+1)}$, we assume that node k is a stationary node and let $w_{ik}^{(n)} = 0$. Before all nodes in the networks are slot synchronized, only the nodes that can't adjust its local slot reference are always remove from computation. When all the nodes in the network are slot synchronized, the adjustment values are equal to zero, namely $\varepsilon_i^{(n)} = 0$.

Secondly, when a new node joins in the network, the time slot references of its neighboring nodes will also change.

Assume that a new coming node entering into the network (with N nodes) at n^{th} adjustment when all other nodes in the network are synchronized at a common time slot reference. That is: $t_i^{(n)} = t$, $(i = 1,2 \cdots N)$

The slot reference of any arbitrary node in the network at the next iteration computation is:

$$t_i^{(n+1)} = \sum_{j=1}^{N+1} w_{ij}^{(n)} t_j^{(n)} = \sum_{j=1}^{N} w_{ij}^{(n)} t^{(n)} + w_{i,N+1} t_{N+1}^{(n)}$$

If $t_{N+1}^{(n)} \neq t$, the $t_i^{(n+1)}$ will change according to $t_{N+1}^{(n)}$, and thus $t_i^{(n+1)} \neq t$. To avoid interference introduce by new coming nodes, Let $w_{i,k}^{(n)} = 0$ if node k is a new coming node at n^{th} computation. Hence, for any arbitrary node in the network:

$$t_i^{(n+1)} = \sum_{j=1}^{N+1} w_{ij}^{(n)} t_j^{(n)} = \sum_{j=1}^{N} w_{ij}^{(n)} t^{(n)} = t.$$

The interference is eliminated from the computation. And for the new coming node, the slot reference of new coming nodes is quickly synchronized to other nodes:

$$t_{N+1}^{(n+1)} = \sum_{j=1}^{N+1} w_{N+1,j}^{(n)} t_j^{(n)} = t$$

Thirdly, a malfunctioning node whose time slot reference is randomly changing, exists in the network, it will lead to non-convergence of the algorithm.

Assume that node k is a vibrating node in the network with N nodes, the time slot reference is un-comply with the slot synchronization algorithm, and keep changing randomly. Let $t_k^{(n)} = \xi^{(n)}$, $\xi^{(n)}$ is a random number. Therefore, the slot reference of any arbitrary node in the network at the next iteration computation is:

$$t_i^{(n+1)} = \sum_{j=1}^{N} w_{ij}^{(n)} t_j^{(n)} = \sum_{j=1, j\neq k}^{N} w_{ij}^{(n)} t_j^{(n)} + w_{i,k}\xi^{(n)}$$

Obviously, $t_i^{(n+1)}$ keep changing according to $t_k^{(n)}$. Consider that all $\Delta t_{ij}^{(n)}$ are no more than a threshold value μ when all nodes in the network comply with the algorithm. If $\Delta t_{ik}^{(n)} > \mu$, node k is assumed as a vibrating node. To avoid interference introduce by vibrating node, let $w_{ik}^{(n)} = 0$, hence:

$$t_i^{(n+1)} = \sum_{j=1, j\neq k}^{N} w_{ij}^{(n)} t_j^{(n)} = t \text{, here } \sum_{j=1, j\neq k}^{N} w_{ij}^{(n)} = 1$$

The vibrating nodes are eliminated from the computation, and its influence is avoided. Simulation results will be illustrated in section 4.

4. SIMULATION RESULTS

We set up our simulation as follows to verify the performance of this weighted parameters:

A random topology always-connected network with 25 nodes is generated within the range of a rounded area. The original time slot reference of each node follows a uniform distribution on the interval of [0,Ts]. T_s is a time slot period. The overcast radius of a node is 0.4 times that of the rounded area.

We simulate this algorithm 3000 times, in each simulation the time slot reference of every node were iteratively adjusted for 50 times. The average of maximal normalized time slot difference and the average of maximal normalized slot adjustment in the 3000 times simulation are computed, and that, their relation curves versus the iteration times are showed in the following. Normalized slot difference is defined as the slot difference divided by the length of a time slot period. Normalized slot adjustment is defined as the adjustment divided by the length of a time slot period.

A. Scenario (1)

All nodes in the network can adjust their time slot reference, and there is no time slot drift, the relation curve is showed in Figure 2. From the figure we see that after 15 times iteration, the average maximal normalized slot differences are all less than 10^{-6}.

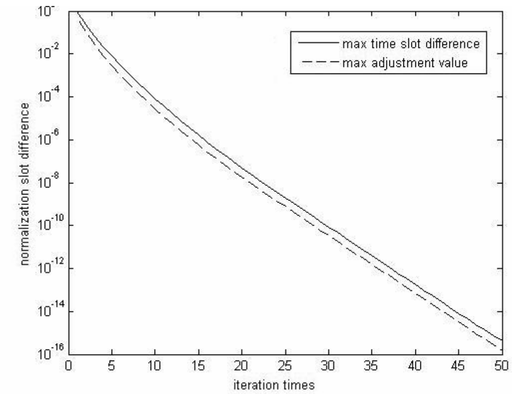

Figure2 convergence curve of scenario (1)

B. Scenario (2)

There are 3 nodes of stationary clock in the network that couldn't adjust its time slot reference, and there is no time slot drift. The simulation result without using anti-interference method is shown in Figure 3. From the graphics we find that the stationary nodes in the network lead to un-convergence in the decentralized slot synchronization algorithm.

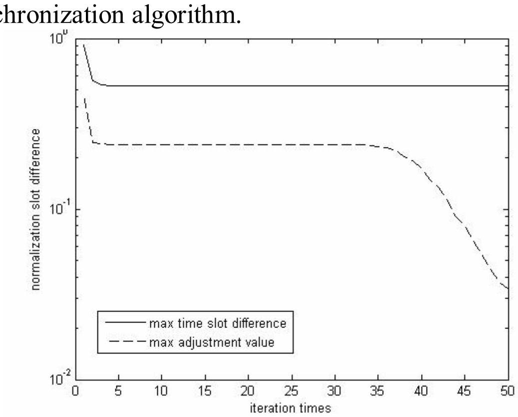

Figure3 convergence curve of scenario (2) without using the method

The simulation result with using anti-interference method is shown in Figure 4. From the graphic we find that the node that can't adjust its time slot reference is eliminated from iteration computing. The convergence rate is the same as that of scenario (1).

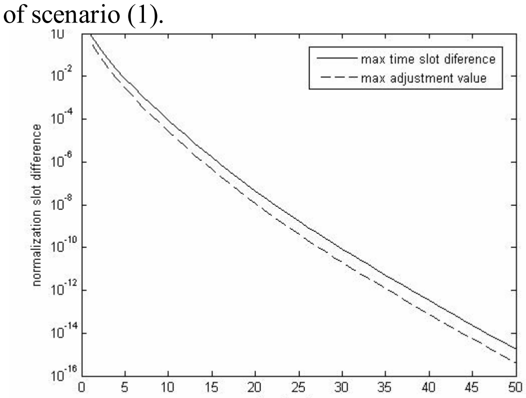

Figure4 convergence curve of scenario (2) with using the method

C. Scenario (3)

There are 2 new joining node coming into the network at the 10^{th} and 20^{th} iteration computing respectively, and there is no time slot drift, the simulation results without using anti-interference method is shown in Figure 5. From the graphics we find that the new joining nodes lead to temporary vibration in time slot reference, after that, it began to re-convergence.

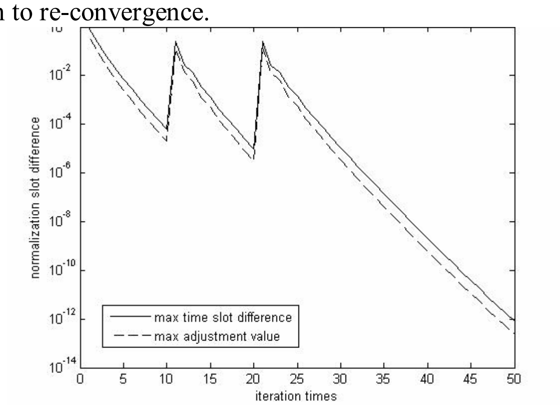

Figure5 convergence curve of scenario (3) without using the method

The simulation result with using anti-interference method is shown in Figure 6. From the graphics we find that the affection of new coming nodes to the decentralized slot synchronization is removed.

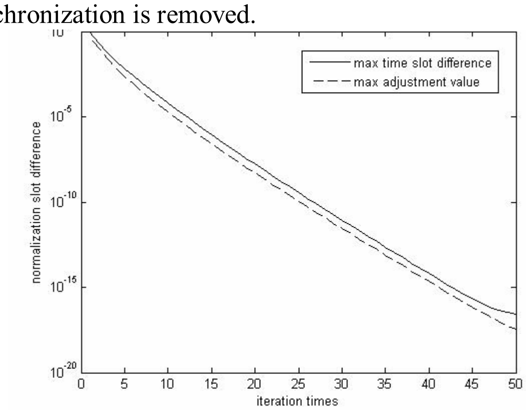

Figure6 convergence curve of scenario (3) with using the method

D. Scenario (4)

There are 2 bouncing node whose time slot reference randomly changes in the range from 0 to T_s. All nodes in the network could adjust its time slot reference, and there is no time slot drift, otherwise, The simulation result without using anti-interference method is shown in Figure 7. The figure shows that the bouncing node in the network leads to un-convergence in the algorithm. After 10 times iteration, the average maximal normalized slot differences are keep in the range of 10^{-1} and 10^{-2}.

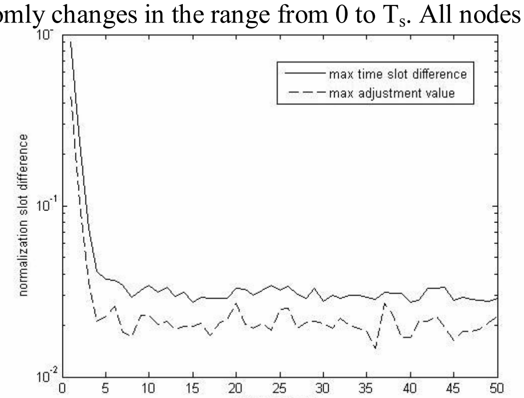

Figure7 convergence curve of scenario (4) without using the method

The simulation result with using anti-interference method is shown in Figure8. From the graphics we find that the nodes that can't adjust its time slot reference are eliminated from iteration computing.

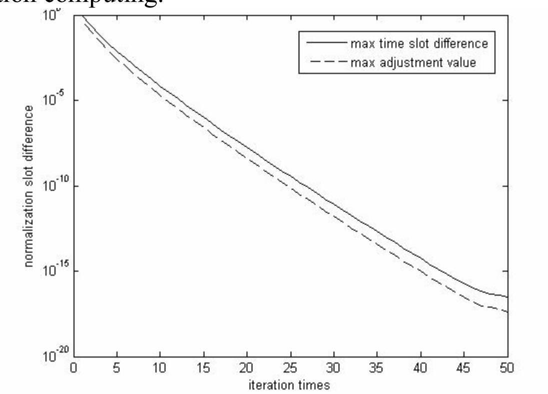

Figure8 convergence curve of scenario (4) with using the method

5. CONCLUSION

In this paper, an interference elimination method for decentralized slot synchronization in TDMA-based wireless ad hoc networks is presented. Factors influencing the convergence of slot synchronization are also evaluated. Simulation results show that this method automatically eliminates the effect of interferential nodes such as stationary nodes, new coming nodes and vibrating nodes. And it achieves high robustness performance.

11. REFERENCES

[1]Henrik Lonn, "Initial Synchronization of TDMA Communication in Distributed Real-Time Systems," Proceedings of the 19th IEEE International Conference on Distributed Computing Systems, pp. 370-379

[2]Essam Souour, M.Nakagawa, "Mutual decentralized synchronization for intervechicle communications," IEEE Trans ion Vehicular Technology, Vol.48, Issue 6, pp.2015-2027

[3]Qi Yang, Jianghong Shi, "A slot synchronization method and performance analysis of TDMA Ad Hoc network," Proceedings of 2007 IEEE International Workshop on Anti-counterfeiting Security, Identification, pp.340-343

[4]André Ebner, Matthias Lott, "Decentralized Slot Synchronization In Highly Dynamic Ad Hoc Networks," Proceedings of the 5th International Symposium on Wireless Personal Multimedia Communications, Vol.2, pp.494-498, 27-30 Oct. 2002

[5]Qi Yang, Jianghong Shi. "Parameter Selection for Decentralized Slot Synchronization of Wireless Ad Networks," The 2nd International Conference on Computer Science & Education, pp.922-925