A Quorum-based Mechanism as an Enhancement to Clock Synchronization Protocols for IEEE 802.11 MANETs

Shu-Min Chen, Sheng-Po Kuo, and Yu-Chee Tseng

***Abstract*— In wireless mobile ad hoc networks (MANETs), it is essential that all mobile hosts (MHs) are synchronized to a common clock to support the *power-saving (PS)* mechanism. Many protocols have been proposed for clock synchronization in IEEE 802.11 MANETs. However, it is practically impossible for any protocol to completely solve the asynchronism problem especially when connectivity is achieved by multi-hop communication or when a network could be temporarily disconnected. In this work, we propose a *quorum-based mechanism*, which includes a new structure of beacon intervals for MHs to detect potential asynchronous neighbors and an enhanced beacon transmission rule to assist clock synchronization protocols to discover asynchronous neighbors within bounded time. The proposed mechanism should be regarded as an enhancement to existing clock synchronization protocols. Our simulation results show that the mechanism can effectively relieve the clock asynchronism problem for IEEE 802.11 MANETs.**

***Index Terms*— Mobile ad hoc network, power saving mechanism, wireless network, clock synchronization.**

I. Introduction

In wireless mobile ad hoc networks (MANETs), it is essential that all mobile hosts (MHs) are synchronized to a common clock to support the *power-saving (PS)* mechanism. In IEEE 802.11 PS mechanism, each MH wakes up at the beginning of a beacon interval to exchange messages. Through message exchanging, MHs can schedule communications for the current beacon interval. If a MH is not scheduled for any communication activity, it can go to the doze mode for the rest of the beacon interval. In order to exchange messages properly, MHs' beacon intervals should be synchronized. Otherwise, PS mechanism will not function well.

To fulfill the requirement of clock synchronization, IEEE 802.11 specifies a distributed *Timing Synchronization Function (TSF)* for ad hoc networks. Since an ad hoc network has no infrastructure to provide a centralized synchronous mechanism, MHs will compete with each other to broadcast their timing information through *beacons*. Each MH receiving beacons will adjust its clock to synchronize with the sender if its current time is slower than the timestamp in the beacon. Note that the clock is unchanged if its current time is faster. The TSF mechanism is quite enough for small and static ad hoc networks. However, it is pointed out in [1] that IEEE 802.11 TSF has the scalability problem. As the number of MHs increases, the contention among beacons may get very serious, causing beacon loss and thus the clock asynchronism problem. A simple protocol called ATSP is then proposed, which gives faster MHs higher priorities to transmit their beacons. Most existing clock synchronization protocols are mainly designed for fully-connected MANETs. For multi-hop MANETs, several protocols, such as *Automatic Self-time-correcting Procedure (ASP)* [2], have been proposed.

It is to be noted that the above protocols all aim at minimizing the maximum clock drift among MHs. If two MHs' wake-up schedules do not overlap with each other, these protocols can not re-synchronize them. In fact, it is practically impossible for any synchronization protocol to completely solve the asynchronism problem especially when connectivity is achieved by multi-hop communication or when a network could be temporarily disconnected. We will give several examples to prove this argument. MHs' mobility may further aggravate the clock asynchronism problem. Unable to find asynchronous neighbors will hurt network connectivity and thus communication performance of a network.

This paper proposes a new quorum-based mechanism which can serve as an enhancement to existing clock synchronization protocols and thus should cooperate with one to relieve the clock asynchronism problem. The concept of quorum is borrowed from [3] to guarantee detection of asynchronous neighbors. However, [3] does not try to synchronize MHs' clocks. Instead, a MH tries to predict other MHs' clocks by keeping their clock differences and thus needs to deliver buffered packets at right time, which is inefficient in terms of network throughput.

II. Problem Definition and Backgrounds

In the PS mechanism of the IEEE 802.11 ad hoc mode, a beacon interval consists of a beacon window, an ATIM window, and a DATA window (Fig. 1(a)). A PS MH only needs to wake up during the beacon window and check delivery requests during the ATIM window. The *clock asynchronism problem* occurs when the time difference between any two neighboring MHs is larger than the length of one beacon window. In Fig. 1(a), A can hear beacons sent by B, but B may not be able to to hear A's beacons and thus get synchronized with A. The clock asynchronism problem may remain unsolved until the amount of time drift between A and B is a multiple of one beacon interval (refer to Fig. 1(b)). To see how serious this problem is, suppose that A is faster than B by $20\mu s$ per beacon interval in Fig. 1. Assuming that the

Manuscript received November 30, 2006. The associate editor coordinating the review of this letter and approving it for publication was Dr. Rohit Nabar.

S. M. Chen and S. P. Kuo are with the Department of Computer Science, National Chiao-Tung University, Taiwan (e-mail: spkuo@cs.nctu.edu.tw).

Y. C. Tseng is with the Department of Computer Science, National Chiao-Tung University, Taiwan, and the Department of Information and Computer Engineering, Chung-Yuan Christian University, Taiwan.

Y. C. Tseng's research is co-sponsored by Taiwan's MoE ATU Program, by NSC under grant numbers 93-2752-E-007-001-PAE, 95-2623-7-009-010-ET, 95-2218-E-009-020, 95-2219-E-009-007, 94-2213-E-009-004, and 94-2219-E-007-009, by Realtek Semiconductor Corp., by MOEA under grant number 94-EC-17-A-04-S1-044, by ITRI, Taiwan, and by Intel Inc.

Digital Object Identifier 10.1109/LCOMM.2007.061963.

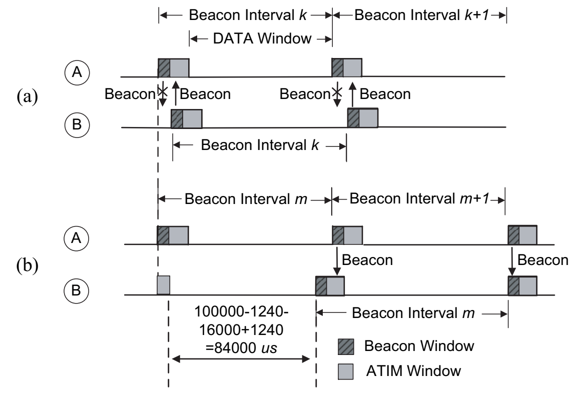


Fig. 1. (a) An example of the clock asynchronism problem when MH B is slower than MH A, and (b) resynchronization after B catches up with A's beacons.

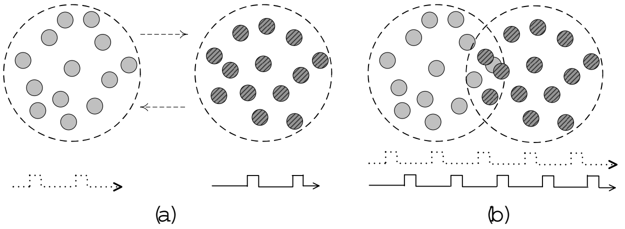


Fig. 2. An example where clock asynchronism occurs when two disconnected components of a MANET meet each other.

lengths of a beacon interval, an ATIM window, and a beacon window are $100000\mu s$, $16000\mu s$, and $1240\mu s$, respectively (based on the IEEE 802.11 DSSS recommendation), it will take $\frac{100000-1240-16000+1240}{20} = 4200$ beacon intervals ($= 420$ seconds) to move from the scenario in Fig. 1(a) to the scenario in Fig. 1(b).

Most existing clock synchronization protocols aim at minimizing the maximum clock drift among MHs in a connected and initially synchronized network. It lacks a mechanism to get MHs synchronized when their clocks have seriously drifted away. In a MANET, mobile MHs may be temporarily partitioned into multiple groups. During this period synchronization between groups is impossible. For example, Fig. 2(a) illustrates two groups of mobile MHs, each being perfectly clock synchronized but mutually asynchronous. When these two groups merge into one, as shown in Fig. 2(b), MHs in these two groups may not be able to discover each other, and thus the network remains disconnected, which is clearly harmful. Even if a MANET remains connected, a small clock drift per hop may accumulate into a large amount of drift after multiple hops. As shown in Fig. 3, if neighboring MHs' clock drift is $\frac{1}{5}$ of one beacon window, MHs A and F, which are separated by 5 hops, may still remain out-of-synchronization. Therefore, when F moves to A's communication range, they may not discover each other, making the network layer mistakenly interpret that A and F are quite far away.

III. The Proposed Quorum-Based Mechanism

Our goal is to design an enhancement that can co-work with existing clock synchronization protocols to relieve the clock asynchronism problem. The basic idea is to use a "quorum" concept to schedule MHs' wake-up time to ensure that a MH can always detect any nearby asynchronous MH within bounded time. Once such MHs are detected, our beacon transmission rule will help synchronize their clocks. The *grid quorum* is first proposed in [4] for a MH to discover asynchronous neighbors. Basically, it requests a PS MH to wake up in only necessary beacon intervals for the discovery purpose. Compared to a random wake-up approach, it can guarantee to discover neighbors within bounded time.

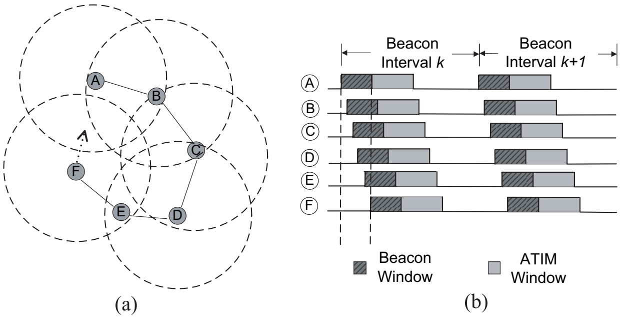


Fig. 3. An example where clock asynchronism causes the network lose the communication link between MHs A and F.

A. Structure of Beacon Intervals

A grid quorum is a 2D $N \times N$ matrix such that each quorum contains a random column and a random row of the matrix. Clearly, the intersection of any two quorums is non-empty. Given a quorum, a MH will group its beacon intervals such that N^2 consecutive beacon intervals constitute one group. In each group, its N^2 beacon intervals are arranged by row-major in an $N \times N$ grid. The $2N-1$ beacon intervals in the selected column and row are *quorum intervals*, and the remaining $N^2 - 2N + 1$ beacon intervals are *non-quorum intervals*.

Each quorum and non-quorum interval is divided into three parts: *beacon window*, *ATIM window*, and *DATA window*. Beacon transmission rules during a beacon window will be defined in the next section. A MH's behaviors during ATIM and DATA windows are the same as those defined in IEEE 802.11 except that it has to stay awake throughout the whole DATA window during a quorum interval. Fig. 4 illustrates an example of a grid quorum and the structures of quorum and non-quorum intervals. With such a structure, it has been proven in [4] that two neighboring MHs always have two chances to hear each other's beacons in every N^2 beacon intervals, no matter how much their clocks drift away. The value of N is tunable. A smaller N would facilitate clock synchronization, while a larger N can save MHs' energy. This will be further evaluated in Section IV.

B. Beacon Transmission Rules

Our quorum-based mechanism has to incorporate with an existing clock synchronization protocol (such as [1], [2]). Let $f_i(n)$ be the beacon transmission decision of MH i made by the adopted clock synchronization protocol for the n-th beacon interval ($f_i(n) = 1$ means "transmit" and $f_i(n) = 0$ means "don't transmit"). The actual beacon transmission is controlled by the *beacon-reception* and the *beacon-window* processes in Fig. 5.

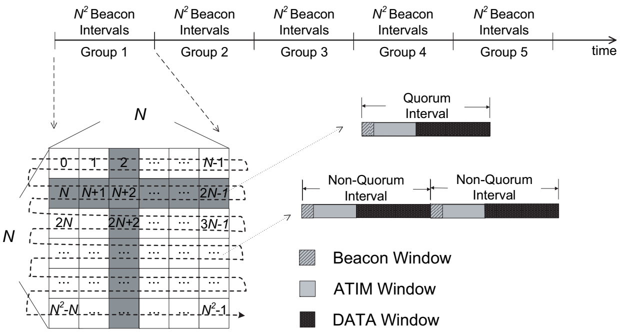
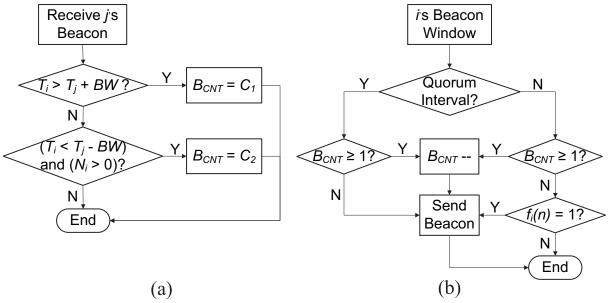


Fig. 4. The structures of grid quorum, quorum interval, and non-quorum interval.



(a) (b)

Fig. 5. (a) Beacon-reception process and (b) beacon-window process for MH i.

The beacon-reception process (Fig. 5(a)) is triggered when MH i receives a beacon from MH j. Let T_i be the current time of i, T_j be the timestamp in j's beacon, BW be the length of a beacon window, and N_i be the number of neighbors of i. MH i will compute beacon counter B_{CNT} as follows:

1) If $T_i > T_j + BW$, this means that i and j are out-of-synchronization. Since j and its neighbors may also remain out-of-synchronization with i, i will set $B_{CNT} = C_1$ to enforce itself to send C_1 beacons in the next C_1 beacon windows.
2) If $T_i < T_j - BW$, i and j are also out-of-synchronization. If i has any neighbor (i.e., $N_i > 0$), it may also be out-of-synchronization with j. So i will set $B_{CNT} = C_2$ to enforce itself to send C_2 beacons in the next C_2 beacon windows.

B_{CNT} is to enforce more beacon transmissions to synchronize potential out-of-synchronization neighbors. We recommend C_1 and C_2 to be set to at least N because an out-of-synchronization MH will enter a quorum interval (and thus stay awake in the whole interval) at least once in the next N beacon intervals. The beacon-window process for MH i (Fig. 5(b)) is triggered when a beacon window arrives. In any of the following events, MH i will try to transmit a beacon.

1) A quorum interval arrives.
2) $B_{CNT} > 0$ (in which case B_{CNT} will be decreased by 1).
3) $f_i(n) = 1$.

IV. Simulations

To verify the effectiveness of the proposed mechanism, a 3000 × 3000 square meters field with 500 random MHs is simulated. Each MH roams around by the random way-point model with a speed of $0 \sim 5$ m/s and a pause time of 20 seconds. The communication range of each MH is identical and is equal to 250 meter. The lengths of a beacon interval, an ATIM window, and a beacon window are 100000 μs, 16000 μs, and 1240 μs, respectively. Each MH's clock speed is uniformly distributed in [$0.9999s$, $1.0001s$], where s is the standard speed.

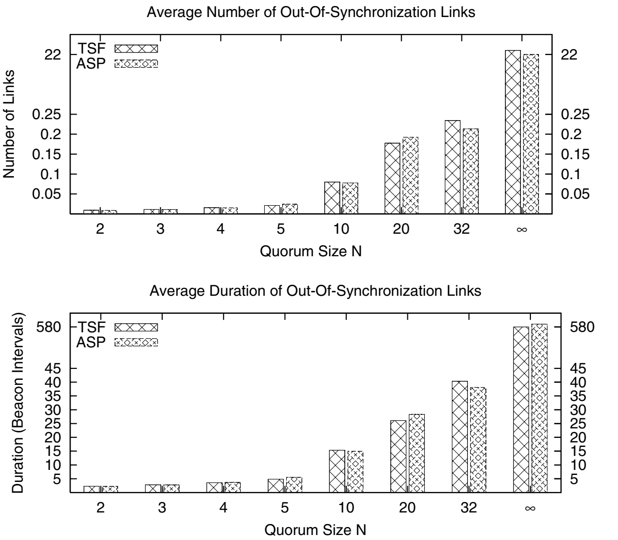


Fig. 6. Comparison of the average number of out-of-synchronization links in a beacon interval and the average duration when an out-of-synchronization link appears under different quorum sizes.

We simulate the standard TSF and the ASP [2] with and without our enhancement. Fig. 6 compares the average number of out-of-synchronization links in a beacon interval and the average duration when an out-of-synchronization link appears under different cases. Note that $N = \infty$ means that no quorum-based mechanism is applied. We can observe that our enhancement can substantially decrease the number of out-of-synchronization links and accelerate the neighbor discovery process.

References

[1] L. Huang and T.-H. Lai, "On the scalability of IEEE 802.11 ad hoc networks," in *Proc. ACM Int'l Symp. on Mobile Ad Hoc Networking and Computing 2002*, pp. 173–182.

[2] J.-P. Sheu, C.-M. Chao, and C.-W. Sun, "A clock synchronization algorithm for multi-hop wireless ad hoc networks," in *Proc. 24th Int'l Conf. on Distributed Computing Systems 2004*, pp. 574–581.

[3] J.-R. Jiang, Y.-C. Tseng, C.-S. Hsu, and T.-H. Lai, "Quorum-based asynchronous power-saving protocols for IEEE 802.11 ad hoc networks," *ACM Mobile Networking and Applications*, vol. 10, no. 1-2, pp. 169–181, 2005.

[4] Y.-C. Tseng, C.-S. Hsu, and T.-Y. Hsieh, "Power-saving protocols for IEEE 802.11-based multi-hop ad hoc networks," in *Proc. IEEE INFOCOM 2002*, vol. 1, pp. 200–209.