2005 International Symposium on Nonlinear
Theory and its Applications (NOLTA2005)
Bruges, Belgium, October 18-21, 2005

Synchronization bottleneck in multi-hop ad hoc networks using 802.11 MAC protocol

Hisa-Aki Tanaka†, Daisuke Ohta†, Akio Hasegawa‡, and Peter Davis‡

†Department of Electronic Engineering, The University of Electro-Communications (UEC)
1-5-1 Choufugaoka, Choufu-shi, Tokyo 182-8585 Japan
‡ATR Adaptive Communications Research Laboratories
2-2-2 Hikaridai, Keihanna Science City, Kyoto 619-0288 Japan

Abstract—The ad hoc mode of IEEE 802.11 is analyzed in the context of node distributions which are typical in widely distributed multi-hop ad hoc networks. We show that when nodes are spread out the transient dynamical behavior of coalescence can be quite complicated, resulting in (i) long times for stabilization of communications and (ii) dead-lock to asynchronous TSF times, in certain situations. Such dynamical behaviours should be considered when designing and implementing communication algorithms, including routing algorithms, at higher layers.

1. Introduction

There is increasing interest in using IEEE 802.11 WLAN cards to set-up ad hoc networks for PC and PDA. Here we discuss a problem with the MAC layer that has been overlooked by researchers working on ad hoc network issues at higher layers; the problem of cell coalescence. The ad hoc mode of IEEE 802.11 supports the formation of independent cells (IBSS) in the absence of an access point. It has a mechanism for autonomous cell creation and association to handle start-up and new nodes joining a cell. However, it was intended primarily for use with compact cells, where all nodes can hear each other. It has not been studied in the context of node distributions which are typical in widely distributed multi-hop ad hoc networks. We show that when nodes are spread out the transient behavior of coalescence can be quite complicated, resulting in (i) long times for stabilization of communications and (ii) dead-lock to asynchronous TSF times, in certain limiting but natural situations.

2. Critical case: Merging of two cells

We claim that coalescence times in multi-hop networks can be characterized using the prototypical case shown in the inset of Fig. 1, which shows the merging of two overlapping cells. We assume that initially they have the same group identity (SSID) and channel but different cell identity (BSSID). This is a situation which often occurs in practice, for example in and around buildings, when nodes move in the intermediate area where they can hear transmissions from both cells.

According to 802.11, when a node hears the beacons from nodes with the same SSID but an later time (TCF) it will change its own BSSID and TCF to match. The fairness of random beacon transmissions should guarantee that eventually all nodes adopt the same BSSID and TCF. However, the transient time depends on the details of the dynamics of beacon generation. Beacons are typically generated within an interval which is only a fraction of the beacon period. When power-saving is adopted, all but one node (the node which last sent a beacon) go to sleep between beacon generation intervals. In the worst case, the beacon generation intervals of the two cells are initially completely out of sync (anti-phased) and each time the intermediate node transmits a beacon only one new node is pulled into coalescence (sync). A rough estimate gives the average time required for coalescence to be of order $2N \times T$, where N is the number of nodes in each cell and T is the beacon interval. For example, for $N = 20$ this could be as long as 40 beacon intervals, or 4 secs for a 100[ms] beacon interval. This means that coalescence times could be longer than the period of Hello packets used in conventional ad hoc routing protocols.

Such estimates can be verified by a rigorous analysis using a probabilistic distribution of coalescence times. In Fig. 1, the rough estimate, rigorous estimate, and direct simulation results [1] of coalescence times are plotted, where they show a perfect agreement. In the rigorous estimate a linear increase of coalescence time holds for all N below a certain critical node number $N^* \sim 80$. Beyond this N^*, coalescence time increases as $(N - N^*)^\beta$ with $\beta \sim 2.5$, which reflects nonlinear increase of beacon collision probability. Details of this analysis will be presented elsewhere.

[1] Here, in the simulation the beacon generation window W is set to 62, and 50000 trials are carried out with random initial conditions of backoff-timer for each node.

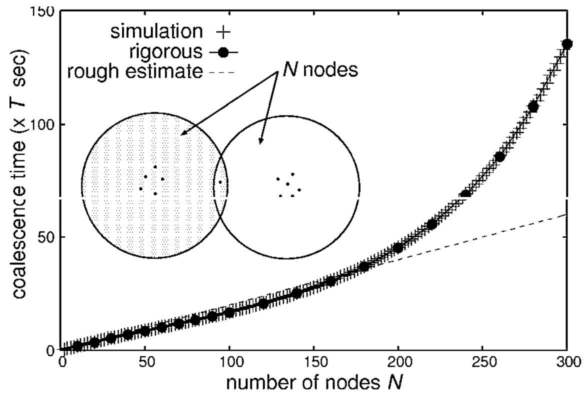

Fig.1 Coalescence time versus node number for the coalescence of two separate compact cells (IBSS) via a node which is within range of both. Beacon generation intervals in the two cells are assumed to be initially out of phase (in anti-phase).

3. Dead-lock to asynchronous states in merging of multiple cells

As opposed to the case of merging independent cells, another consideration is required for a larger spatially extended cell, where its size exceeds the transmission range of each node as shown in the inset of Fig. 2. For this case, if a node with the same SSID but a later time (TCF) is moved into the cell from another one, its BSSID and TCF is propagated among other nodes and eventually the global sync (coalescence) is expected.

However, such coalescence process should be more complicated because multiple coalescence pathways exist at the same time. To analyze this dynamical process, we employed the same simulation protocol of the merging two-cell case, to randomly distributed N nodes as shown in the inset of Fig. 2. Here, a newly introduced node is positioned at random in the cell for each simulation. As in the merging two-cell case, beacon generation intervals of the cell and a newly joined node are set to be anti-phased, to consider the worst situation.

Then, our simulations lead to the following observations:
(i) As shown in Fig. 2(a) propagation of the later TSF time (time updates) proceeds quickly among sparcely distributed nodes or nodes near the boundary of the cell, and then time updates slowly proceed for nodes around the center of the cell.
(ii) Such separation of fast and slow propagation is enhansed as the beacon packet length increases, and as shown in Fig. 2(b) the coalescence time becomes quite long ($\sim$ 10s) if the packet length is over 21 slots.
(iii) Furthermore, if the packet length is not less than the beacon generation window (31 slots in this case),

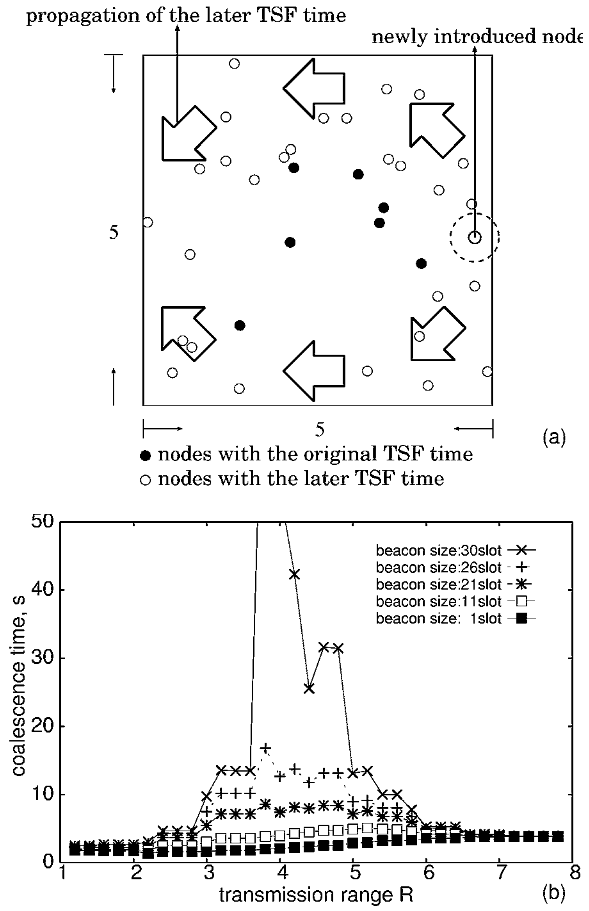

Fig.2 (a) Process of TSF time updates in a larger cell with randomly distributed nodes. The arrow ($\Leftarrow$) shows the propagation direction of TSF time updates. (b) Coalescence time versus transmission range R of each node for five different beacon packet length.

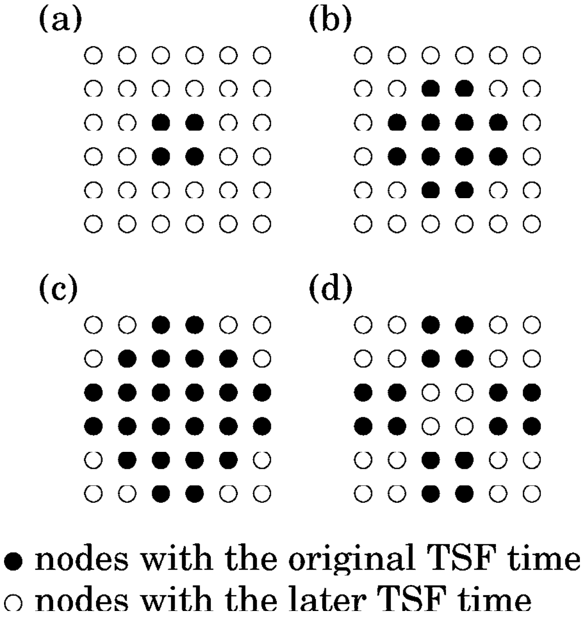
(a)
(b)
(c)
(d)
● nodes with the original TSF time
○ nodes with the later TSF time

Fig.3 Dead-lock patterns of TSF timers obtained for $N = 6^2$.

cases not converging to global coalescence are observed, depending on the choice of N, R and the distribution of nodes. Detailed analysis for these cases shows that a certain group (or groups) of nodes keep the original TSF time and their time cannot be updated from other nodes due to constant collisions of beacon packets.

To elucidate the essential mechanism of this dead-lock generation, we made systematic simulations for the array of nodes (as opposed to the original randomly positioned nodes). In the simulation, we fix the normalized distance between adjacent nodes to 1 and systematically change N from $6^2(= 36)$, 49, ..., to 100. For each array defined above, the (normalized) transmission range R is also systematically changed from 1 to $\sqrt{N}$ (the edge length of the array). In this way, the size N and the range R dependency of coalescence time can be analyzed.

The insets (a),(b),(c), and (d) in Fig.3 are examples of these pattern, which are obtained for $N = 6^2$ and (a) $R = 3.8, 4.0, 4.4, 4.6, 4.8$, (b) $R = 5.0, 5.2, (c)R = 5.4, 5.6, 5.8, 6.0, 6.2, 6.4, 6.6$, and (d) $R = 3.0, 3.2, 3.4$, respectively[2] .

The reason why these patterns (a),(b), and (c) block the global coalescence process can be explained by the fact that the four nodes at the corner of the array have been already synchronized but independently transmitting beacons simultaneously as they are out of their transmission ranges each other.

The pattern (d) in Fig. 3 has a somewhat different structure. However, the mechanism of blocking global coalescence is the same to patterns (a), (b), and (c), and this can be proved easily. Although we have focused on the case of $N = 6^2$ in the above analysis, other patterns found for larger N are turned out to have the same mechanism in the above cases.

In the above analysis, the power-saving mode, namely awake-sleep cycles is assumed, which is an important factor of designing sensor networks. Here, we consider how such power-saving mode affects the dead-lock pattern generation. For this purpose we performed the same simulation for the N nodes array ($N = 36, \ldots, 100$) as well as the randomly positioned nodes but with no awake-sleep cycles; we assumed all nodes are always awake, and 10^6 trials are made in this simulation. The result shows that all of dead-lock patterns observed in the above power-saving case cannot be observed. (However, certain exceptional dead-lock patterns are observed with 0.1% probability.)

As opposed to the random initial condition in the simulation, we set the dead-lock patterns (a-d) as the initial condition, respectively. For these cases, all patterns are maintained and keep blocking the global coalescence. Thus, the power-saving mode is supposed to be an essential factor in self-organizing dead-lock patterns, although it is important in designing adhoc networks.

4. Discussion

The details of convergence depend on many factors, including the topology and the initial distribution of TCF, and dispersion of clock frequencies, the effect of collisions and other packets errors, which could cause nodes to miss beacons resulting in even longer coalescence times. Moreover the coalescence time will be larger for larger networks. For example, if the network consists of a chain of M clusters, then the convergence time may increase by factor of M . However, the prototypical case presented in Fig. 1 provides an effective basis for estimating the coalescence times in more complicated cases.

For the spatially extended case of Fig. 2, multiple coalescence pathways make the dynamics more complex. In the worst case, dead-lock patterns are self-organized which means 802.11 MAC protocol does not function properly.

The need to dynamically form and sustain cells at the MAC level is a major difference from wired networks. Our analysis shows that these dynamics should be considered when designing and implementing communication algorithms, including routing algorithms, at higher layers.

[2] The probability of generating these patterns is obtained as (a) 85.4, 85.4, 55.4, 50.2, 50.0 %, (b) 13.7, 9.8 %, (c) 4.3, 4.3, 1.8, 0.12, 0.26, 0.20, 0.01 %, and (d) 0.4, 0.3, 0.4 %, respectively for 50000 trials.

Acknowledgments

This work was partially supported by NICT and the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Young Scientists (B), 16760292 (2004), and stimulated by the activities of the AdHoc Wireless Network Consortium.

References

[1] IEEE Std. 802.11 Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specification,
http://standards.ieee.org/getieee802/download/802.11-1999.pdf