Vol.2010-MBL-55 No.14
2010/9/3

A Smart Channel Selection Strategy for Multi-rate Multi-channel MAC Protocol

Bin Han[†1] and Ken'ichi Kawanishi[†1]

The IEEE 802.11 wireless media standard supports multiple transmission rates and frequency channels at the physical layer. By using the multi-rate multi-channel Medium Access Control (MAC) protocol, it can increase the performance of ad hoc networks. However, there is another task for multi-rate multi-channel MAC protocol. That is the channel selection strategy. In this paper we introduce a multi-rate multi-channel MAC protocol with one interface using a smart channel selection strategy. The core idea of our channel selection strategy has two main parts. One is to allow the receiver to select the appropriate rate for data packet transmission during the RTS/CTS exchange. The other part is to allow the receiver to decide a suitable channel selection strategy that is to change the channel after the amount of data packets reaches a threshold. Our channel selection strategy can solve two difficult problems: which transmission rate and channel are used for data packet transmission between source and destinations, and how long transmission rate and channel should be used between the same pair of transmission? We show through computer simulation that our channel selection strategy can provide the lower number of packet losses and the higher throughput of ad hoc network.

1. Introduction

The IEEE Medium Access Control (MAC) protocol provides a physical-layer multi-rate and multi-channel capability[1]. For example, IEEE 802.11b sets four possible transmission rates (1, 2, 5.5, and 11 Mbps) and provides three non-overlapping channels simultaneously.

The multi-rate MAC protocol can exploit multiple rates for transmission. By using it, the transmitter can send data packets by a higher transmission rate than the base rate if channel condition is good. Therefore, multi-rate MAC protocols[2)–4)] can improve the performance of networks throughput.

On the other hand, the multi-channel MAC protocol can exploit multiple frequency channels for transmission. By using it, a pair of source and destination nodes chooses one channel from multiple channels and exchanges data packets on the selected channel that is not used by other pairs. That makes concurrent data transmissions without interference with each other. Therefore, multi-channel MAC protocols[5)–16)] have a potential to improve the performance of network throughput significantly.

Furthermore, the multi-rate multi-channel MAC protocols[17),18)] have been proposed and investigated to design medium access mechanism. However, they do not investigate the channel selection strategy for the multi-rate multi-channel MAC protocol. More specifically, the problem is which transmission rate and channel are used for data transmission between source and destinations, and how long transmission rate and channel should be used between the same pair of transmission?

In this paper, we design a multi-rate multi-channel MAC protocol that only uses one interface per node. We mainly design a smart channel selection strategy for our multi-rate multi-channel MAC protocol so that it can select a suitable rate and channel at a suitable time. Briefly, the core idea of our channel selection strategy has two main parts. One is to allow the receiver to select the appropriate rate for data packet through exchanging control messages in MAC protocol. The other part is to allow the receiver to decide a suitable channel selection strategy that can change the channel after the amount of data packets reaches a threshold.

There are two main contributions in this paper.

- We present a multi-rate multi-channel MAC protocol.
- We design a smart channel selection strategy for multi-rate multi-channel MAC protocol. By using our channel selection strategy, a pair of source and destination of transmission can clearly know which transmission rate and channel should be used. They also can know how long transmission rate and channel should be used.

We use computer simulation to show our multi-rate multi-channel MAC protocol exploiting our channel selection strategy can provide the lower number of packet losses and the higher throughput than other MAC protocols.

†1 Department of Computer Science, Gunma University

2. Related work

2.1 MAC protocol

The basic medium access mechanism in this paper is IEEE 802.11 MAC protocol that uses Distributed Coordination Function (DCF) as physical carrier sensing mechanism. DCF is optional and is known as a four-way handshaking technique. The transmitter uses a Request-to-Send (RTS) message to inform the receiver and to reserve the channel. The receiver shall then reply by acknowledging with a Clear-to-Send (CTS) message. Any other node that overhears the RTS/CTS exchange knows to defer access to the wireless channel for the Network Allocation Vector (NAV) time duration. Once the NAV elapses, the channel is available for access again by any node. Fig. 1 shows the cycle of IEEE 802.11 for a successful transmission.

2.2 Multi-rate MAC protocol

A multi-rate MAC protocol extends the IEEE 802.11 MAC to use multiple physical-layer rates. Such as IEEE 802.11b, it can send data packet at 1 Mbps, 2 Mbps, 5.5 Mbps, and 11 Mbps. In multi-rate MAC protocol, the transmitter can send data packets by a higher rate than the base rate if channel condition is good.

Auto Rate Fallback (ARF)[2] is a MAC protocol that exploits multi-rate capability. In ARF, senders use the error rate of previous transmission to adaptively select future transmission rate. That is, the transmitter will modify its modulation scheme to increase transmission rate in a number of consecutive successful transmissions. Similarly, the transmitter will reduce the transmission rate in consecutive losses. Consequently, if the transmitter has a perpetually high-quality channel, the transmitter will eventually transmit at higher transmission rates.

Receiver-Based Auto Rate (RBAR)[3] is another MAC protocol that exploits multi-rate capability. The key idea of RBAR is to control the transmission rate for receivers. To guarantee that all stations receive the messages (RTS, CTS, and ACK) with error-free, all messages in IEEE 802.11 must be sent at the base rate. Using the received RTS, the receiver determines the maximum possible transmission rate for a given acceptable bit error rate. The receiver sends back the calculated rate to the transmitter by adding it into a special field of CTS. Note

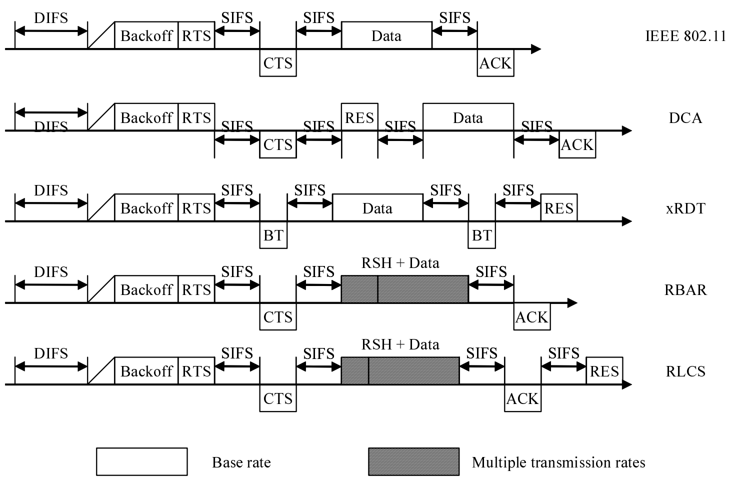

Fig. 1 The medium access mechanism for IEEE 802.11, RBAR, DCA, xRDT, and RLCS.

that all other nodes overhearing this message are also informed of the modified transmission rate. This message is termed as reservation-subheader (RSH) and is inserted preceding data transmission. With the RSH message, overhearing nodes can modify their NAV values to the new potentially decreased transmission time. Fig. 1 shows the cycle of RBAR for a successful transmission. The explicit messaging in RBAR causes a quick adaptation to channel variations and extracts significant throughput gains as compared to ARF.

An extended work of RBAR is Opportunistic Auto Rate (OAR)[4]. The key idea of OAR is to allow transmitter to hold a higher transmission rate on the channel for data transmission. In other words, OAR prefers to use the higher transmission rate to do data transmission than the lower transmission rate.

2.3 Multi-channel MAC protocol

A multi-channel MAC protocol extends the IEEE 802.11 MAC to use multiple frequency channels at physical layer. Such as IEEE 802.11b, it provides three non-overlapping channels. In multi-channel MAC protocol, the transmitter and the receiver can select a suitable channel to transmit data packets without collision. We categorize the MAC protocol using multiple channels based on the node to access multiple channels as follows: *Multiple interfaces*, a node that can access multiple channels simultaneously, and *Single interface*, a node that can only access one channel at a time. Note that the interface is capable of switching from one channel to another.

An approach of multi-channel MAC protocols using multiple interfaces is Dynamic Channel Allocation (DCA)[6] protocol. In DCA, each node has two interfaces, so that it can listen on both the control and data channels simultaneously. The control messages of RTS and CTS are exchanged on the control channel, and data packets are transmitted on the data channel. The similar multi-channel MAC protocols like DCA are proposed[7)–12)]. Comparing to all of those multi-channel MAC protocols, xRDT[13] is a multi-channel MAC protocol based on busy tones. In xRDT, two interfaces are assumed; one is a packet interface used for data transmission and the other is a tone interface for busy tone. xRDT assumes that all of channels have the same bandwidth. Each channel bandwidth is separated into one wide bandwidth for transmission of data packets and one narrow bandwidth for busy tone. A different busy tone is associated for each channel. Therefore, xRDT using two interfaces accesses one channel at a time. It is considered that tone interfaces are simple to implement so that each node in xRDT must have only an additional tone interface, rather than two packet interfaces as in DCA. However, using multiple interfaces increases the cost and complexity of the wireless devices. Fig. 1 shows the cycle of DCA and xRDT for a successful transmission.

An approach of multi-channel MAC protocols using single interface is Multi-channel MAC (MMAC)[14] protocol. This protocol does not need a separate control channel. Instead, it utilizes an ad hoc Traffic Indication Message (ATIM) window in the common channel to negotiate channels using one interface. The ATIM window is the time synchronization phase when 802.11 Power Saving Mechanism (PSM) is applied. The extended work of MMAC to improve the energy efficient of MMAC are proposed[15),16)]. Although MMAC requires only one interface at each node, the overhead on the common channel affects the capacity of ad hoc network when the duration of ATIM window has to be long enough to accommodate all nodes in the neighborhood. Furthermore, global timing issues add to the complexity and it is well known that synchronous scheme is not suitable for mobile networks.

2.4 Multi-rate Multi-channel MAC protocols

Multi-channel Opportunistic Auto Rate (MOAR)[17] is a MAC protocol to support multiple rates and channels for ad hoc networks. MOAR is an extended work of OAR. The key idea of MOAR is to allow nodes to find the band with the best channel quality. Since MOAR defines a *home channel* for exchanging RTS/CTS like the control channel of DCA, one drawback of MOAR is the collision of packets happens easily on the home channel. Furthermore, the home channel may become to a bottleneck. This is because when the number of channels and transmission flows are large, the control channel is filled with all the negotiation of messages and too much contention will cause bottleneck and saturation on the control channel[14].

Opportunistic Multi radio MAC (OMMAC)[18] is also a MAC protocol to support multiple rates and channels for ad hoc networks. The key idea of OMMAC is to collect the physical layer feedback over multiple radios simultaneously and schedule multiple transmissions on available channel accordingly. However, using multiple interfaces increases the cost and complexity of the wireless devices, and may not be suggested for small devices, such as sensor node. For these reasons, we study each node has one interface capable on multi-channel MAC protocol in this paper.

Both of these multi-rate multi-channel MAC protocols do not consider the channel selection strategy. Especially, they do not investigate which transmission rate and channel are used for data transmission between source and destinations, and how long transmission rate and channel should be used between the same pair of transmission. By carefully selecting data channels, the collision of packets may be alleviated. Furthermore, the transmitter can send data packets by transmission rate higher than the base rate if channel condition is good. Therefore, wireless resources may be used more efficiently under carefully designed channel selection strategy, leading to improve the performance of ad hoc networks. The channel selection approach is the main focus in this paper.

3. The proposed multi-rate multi-channel MAC protocol

In this section, we have two subsections. We describe our multi-rate multi-channel MAC protocol briefly at first. Then, we will mainly introduce our channel selection strategy for our multi-rate multi-channel MAC protocol. The core idea of our channel selection strategy has two main parts. One is to allow the receiver to select the appropriate rate for data packet during the RTS/CTS exchange. The other part is to allow the receiver to decide a suitable channel selection strategy that is to change the channel after the amount of data packets reaches a threshold. Our multi-rate multi-channel MAC protocol is based on RBAR. Since the first part was presented in RBAR[3], we will mainly introduce the second task in this paper.

In our protocol, the receiver can decide which rate or channel will be used to send next data packets. Comparing to RBAR, our MAC protocol has the following differences:

- The total available bandwidth W is divided into M non-overlapping frequency channels.
- Nodes are equipped a switchable interface. A node can either receive or transmit packets at a time but not both simultaneously.
- Nodes can sense carrier on all channels when they are idle. Each node keeps a *channel table*.
- Our protocol adds a new control message, which is reservation (RES) message. We use this control message to reserve a new channel between the transmitter and the receiver.
- All of control messages (RTS, CTS, ACK, and RES) format should be extended for adding channel information.

To design a smart channel selection strategy for multi-rate multi-channel MAC protocols that enables the transmission pair to select a suitable rate and channel at a suitable time, we need to solve four difficult questions. They are:

- How does each node find out a new channel?
- How does each node inform neighboring nodes to changing its channel?
- Which transmission rate and channel should be used?
- How long transmission rate and channel should be used?

To find out the new channel of the receiver, the transmitter keeps on scanning channels using RTS until the transmitter can get CTS from the receiver. After RTS was broadcast, if the transmitter did not receive CTS from the receiver, it switches to the next channel to broadcast RTS. By next channel, we mean the channel that is next to the current channel in cyclic order, such as *(current channel + 1)* modulo M. When the transmitter gets CTS from the receiver on channel ch, the transmitter uses channel ch to send the data packet.

After the receiver gets the data packet, it uses a channel selection algorithm to find a new channel from available channels in its own channel table. We will introduce our channel selection algorithm in next subsection. After the receiver decides a new channel, the receiver sends RES that includes the selected new channel to the transmitter on the current channel. When the transmitter gets RES, it has two changes; one is switching the on-going channel to the new channel based on the RES message if it still has data packet to send, and the other one is looking up the new channel if the transmitter needs to send the data packet to another the receiver.

4. The key channel selection strategy

It is intuitively obvious that increasing the concurrency of data transmissions in our multi-rate multi-channel MAC protocol heavily depends on how efficiently the channel is assigned to each of nodes. We propose a dynamical algorithm of selecting channels. The basic idea of our strategy is to change the channel after the amount of data packets reaches a threshold[19].

Suppose that there are $M > 1$ available channels. For each channel, we denote by th_{ch} the threshold for channel c. We assume th_{ch} is pre-determined and a constant which may be different between channel-to-channel. We also introduce a time-dependent variable $r_c(t)$ for channel c as the cumulative number of data packets that the receiver received on the current channel. It is assumed that the receiver can monitor and count the number of data packets received.

At the time when the channel should be changed, the receiver sets $r_c(t) = 0$, and then starts to measure $r_c(t)$ until a new channel is selected. The on-going channel c is changed by using channel selection strategy when $r_c(t)$ is equal to or exceeds the threshold th_{ch}. For example, the channel is updated every time the receiver received one data packet if the threshold is equal to one.

We consider two channel selection strategies:

- *Conditionally Random Channel Selection* (CRCS) strategy chooses the channel in "conditionally random" way: a new channel is selected from available channels other than the on-going channel at random for its own channel table in each node. For example, if the on-going channel is channel 1, then remaining $M-1$ channels (channels 2 through M) are equally selected as a new channel with probability $1/(M-1)$.

- *Longest Idle Channel Selection* (LCS) strategy chooses the "longest idle" channel: each node selects the channel of the longest idle one from its own channel table. Note that we assume the longest idle channel means the channel that has the longest NAV time duration.

We summarize CRCS as follow:

send data packets to the receiver
$th_{ch} = 1$
$r_{ch}(t) = 0$
while $r_{ch}(t) < th_{ch}$ **do**
    update $r_{ch}(t)$ when a new data packet is received
end while
go into CRCS strategy to find new channel

We also summarize LCS as follow:

send data packets to the receiver
th_{ch} = buffer size
$r_{ch}(t) = 0$
while $r_{ch}(t) < th_{ch}$ **do**
    update $r_{ch}(t)$ when a new data packet is received
end while
go into LCS strategy to find new channel

Our channel selection strategy is based on the two channel selection strategies and designed so that it can select a suitable rate and channel at a suitable time for data transmission. The core idea to do that is using Longest Idle Channel Selection if the channel condition is good and using Conditionally Random Channel Selection if the channel condition is not good. By switching the channel selection strategies in this way, we expect that data packet can be hold on a good channel as many as possible when channel condition is good, and data packet can be found in a good channel as soon as possible when channel condition is not good. Since we hybrid Conditionally Random and Longest Idle Channel Selection strategies in our channel selection strategy, we call our channel selection strategy RLCS.

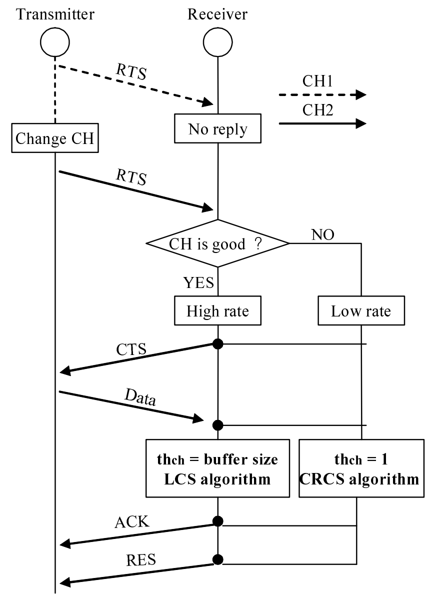

Fig. 2 The process of our channel seleciton algorithm.

Figure 2 shows the process of our channel selection strategy. The transmitter sends RTS to the receiver. If it does not get reply from the receiver, it changes another channel to send RTS. After the receiver gets RTS, it uses Signal-to-Noise Ratio (SNR) to determine the channel condition (for more detail see[3)]). This information is sent by CTS from the receiver to the transmitter. If the channel condition was good, the receiver uses high rate to receive data packet and goes into LCS, while setting th_c = buffer size. On the other hand, if the channel condition was not good, the receiver uses low rate to receive data packet and goes into CRCS, while setting $th_c = 1$. After transmitting data packet, the receiver sends ACK to the transmitter. Then, it still uses the current channel to send RES that includes the new channel information to the transmitter. In

Fig. 1, we show the cycle of RLCS for a successful transmission.

By using RLCS, we can know which transmission rate and channel can be used between transmission pairs. By using SNR and threshold, we can also know how long transmission rate and channel should be used between transmission pairs. We expect that our channel selection strategy can reduce the collision of packets, leading to improve the performance of ad hoc network.

We believe that RLCS has some advantages. First, it improves and balances utilization across all available channels. Secondly, it can also be said that the algorithm is traffic adaptive so channels are changed adaptively, not fixed on one channel, according to the offered load.

5. Performance evaluation

In this section, we use *ns2*[20] to evaluate the performance of our channel selection strategy–RLCS. We compare the proposed MAC protocol with Receiver-Based Auto Rate (RBAR)[3], Dynamic Channel Allocation (DCA)[6], and Extended Receiver Directed Transmission (xRDT)[13]. We do not compare with Multi-channel MAC (MMAC)[14], since xRDT has shown that it can gain the better performance than MMAC.

We show the parameters of simulation in Table 1. In the graphs, the curves labeled as "RBAR", "DCA" and "xRDT" indicate RBAR, DCA, and xRDT MAC protocols. The curves labeled as "RLCS" indicates our multi-rate multi-channel MAC protocol by using hybrid Conditionally Random and Longest Idle Channel Selection strategies.

In our simulation, the number of nodes we used is 36 nodes. For each scenario, we randomly select half of the nodes as sources and the others as destinations. We use the packet arrival rate of CBR flows to vary offered load. We examine the number of packet losses and the throughput with different MAC protocols.

We use two metrics to evaluate the performance of RBAR, DCA, xRDT, and RLCS.

- *Packet losses over all flows in the network:* The packet collision can cause packet losses. In our simulation, we define *losses* include control messages and data packets losses. We compute the sum of the number of losses over all nodes.

Table 1 The parameters of simulation

Parameters	Values
Default transmission rate	2 Mbps
Multiple transmission rate	1, 2, 5.5, 11 Mbps
Number of channels	3
Channel model	Two-ray ground
Transmission range	250 m
Carrier sensing range	550 m
Slot Time	20 μs
DIFS	50 μs
SIFS	10 μs
CW_{min}	32 Slots
CW_{max}	1024 Slots
RTS for single IEEE 802.11	44 bytes
CTS for single IEEE 802.11	38 bytes
ACK for single IEEE 802.11	38 byets
RTS for multiple chs protocol	44 + 2 = 46 bytes
CTS for multiple chs protocol	38 + 2= 40 bytes
RES for multiple chs protocol	38 + 2 = 40 bytes
ACK for multiple chs protocol	38 + 2 = 40 bytes
Busy Tone	1 bytes
Data packet size	1000 bytes

- *Average aggregate throughput over all flows in the network:* RLCS is expected to increase the throughput of network by exploiting multiple rates and multiple channels.

In Fig. 3, we show the number of losses in RBAR, DCA, xRDT, and RLCS. When the network load is low, the number of losses in DCA, xRDT, and RLCS is less than RBAR. This is because they can exploit multiple non-overlapping channels to reduce the collision between transmission pairs, leading to reduce losses. However, when the network load is high, MAC protocols that use only multiple channels cannot reduce losses except RLCS. In general, a node is busy transmitting or receiving on one channel when a neighboring node initiates a channel reservation handshake. Because a node is active on one channel, it is unable to learn of the channel that is selected by its neighboring node and, in turn, may choose the same channel when it begins its next packet exchange. Therefore, the collision still happens in multi-channel ad hoc network. Specially, when the network load is high, this kind of loss greatly reduces the performance. xRDT[13] can alleviate the collision by using busy tone. This is because any

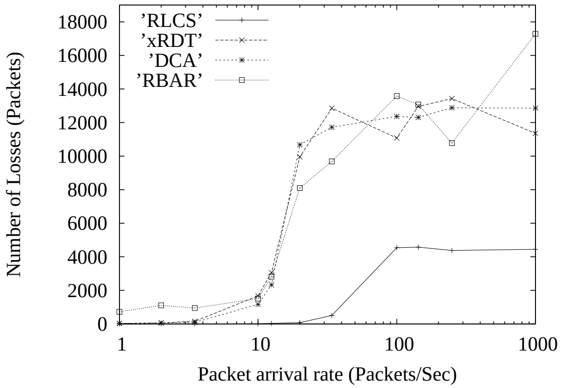

Fig. 3 The number of losses in the RBAR, DCA, xRDT, and RLCS.

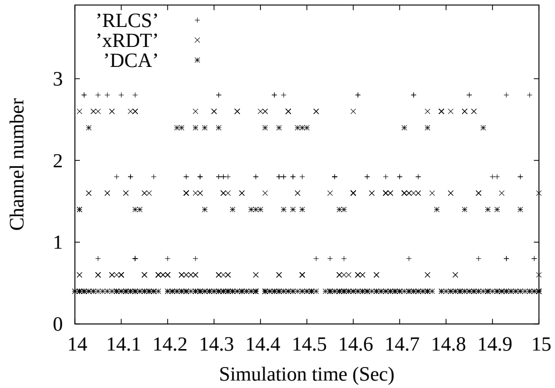

Fig. 4 Losses happen at different channel in RLCS, xRDT, and DCA. Here, we assume channel one is 0–1, channel two is 1–2 and channel three is 2–3.

potential transmitters can defer their transmission by overhearing busy tone. As shown in Fig. 4, we can find the mass of losses happen on the control channel in DCA. Since the potential transmitters can be deferred in xRDT, the loss is reduced and separated over all of channels comparing to DCA. However, as shown in Fig. 3, exploiting busy tone is not effective to reduce losses when the network load is high.

The cycle of RLCS for a successful transmission is similar with that of xRDT as show in Fig. 1. However, we can find the number of losses in RLCS is obviously less than xRDT regardless of the low or high load network in Fig. 3. We can also find RLCS can control packet losses effectively comparing to DCA and xRDT as shown in Fig 4. The prime reason is our channel selection strategy. Since multi-rate multi-channel MAC protocol by using RLCS provides an effective solution for the problems of which channel and when to change a new channel for each node, each node can select a suitable channel at the suitable time for transmission. By carefully selecting data channels, the hidden terminal problem can be alleviated, leading to reduce the number of losses.

In Fig. 5, we show the average aggregate throughput in RBAR, DCA, xRDT, and RLCS. When the network load is low, all of MAC protocols present a similar performance. However, when the network load is high, RLCS can gain higher throughput than other MAC protocols obviously. This is because RLCS can select a suitable channel at the suitable time to send data packets, which reduces packet losses. Furthermore, RLCS enables the transmitter to use a higher transmission rate than the base rate if channel condition is good.

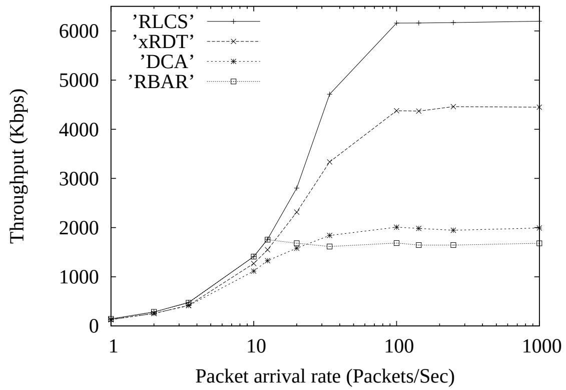

Fig. 5 Average aggregate throughput in the RBAR, DCA, xRDT, and RLCS.

6. Conclusion and future work

This paper mainly has presented a smart channel selection strategy for multi-rate multi-channel MAC protocol. The core idea of our channel selection strategy has two main parts. One is to allow the receiver to select the appropriate rate for data packet during the RTS/CTS exchange. The other is to decide a suitable channel selection strategy that changes the channel after the amount of data packets reaches a threshold. By using our channel selection strategy, our multi-rate multi-channel MAC protocol can select a suitable rate and channel at the suitable time to send data packets. Our channel selection strategy can reduce the hidden terminal problem, which leads to improve the throughput and to reduce the packet loss of ad hoc networks.

Future work will consider larger and more realistic network topologies, which have more channel models such as Rayleigh and Ricean fading environments and have even greater channel contention such as node mobility.

References

1) B.-O.Hara and A.Petrick, "IEEE 802.11 handbook," A Designer's Companion, 1999.
2) A.Kamerman and L.Monteban, "WaveLAN II: A high-performance wireless LAN for the unlicensed band," Bell Labs Technical Journal, pp.118-133, Summer 1997.
3) G. Holland, N. Vaidya, and P. Bahl, "A rate-daptive MAC protocol for multi-hop wireless networks," In Proc. 7th Annual International Conference on Mobile Computing and Networking, pp.236-251, 2001.
4) B.Sadeghi, V.Kanodia, A.Sabharwal, and E.Knightly, "Opportunistic media access for multirate ad hoc networks," In Proc. ACM International Symposium on Mobile Ad Hoc Networking and Computing, pp.24–35, 2002.
5) N.Shacham and J.-B.King, "Architectures and performance of multichannel multihop packet radio networks," IEEE Journal on Selected Areas in Communications, vol.sac-5, No.6, pp.1013–1025, July 1987.
6) S.-L.Wu, C.-Y.Lin, Y.-C.Seng, and J.-P.Sheu, "A new multi-channel MAC protocol with on-demand channel assignment for multi-hop mobile ad hoc network," Proc. 2000 International Symposium on Parallel Architectures, Algorithms and Networks, pp.232–237, Dec. 2000.
7) A.Nasipuri, and S.-R.Das, "Multi-channel CSMA with signal power-based channel selection for multi-hop wireless networks," Proc. IEEE Vehicular Technology Conference, Sep. 2000.
8) T. Luo, M.Motani, and V.Srinivasan, "CAM-MAC: A cooperative asynchronous multi-channel MAC protocol for ad hoc networks," Proc. 2006 Broadband Communications, Networks and Systems, pp.1–10, Oct. 2006.
9) J.-P.Shi, T.Salonidis, and E.-W.Knightly, "Starvation mitigation through multi-channel coordination in CSMA multi-hop wireless networks," Proc. 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing, pp.214–225, May 2006.
10) R.-S.Huang, H.-Q.Zhai, C. Zhang, and Y.-G.Fang, "SAM-MAC: An efficient channel assignment scheme for multi-channel ad hoc Networks," Computer Networks, vol.52, pp.1634–1646, June 2008.
11) P.-J. Wu and C.-N. Lee, "Connection-oriented multi-channel MAC protocol for ad-hoc networks," Computer Communications, vol.32, pp.169–178, 2009.
12) C.-W.Chen, C.-C.Weng, and C.-J.Ku, "An overlapping and pipelining data transmission MAC protocol," Parallel Computing, vol.35, pp.313-330, 2009.
13) R.Maheshwari, H.Gupta, and S.R.Das, "Multichannel MAC protocols for wireless networks," Proc. 3rd Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, vol.2, pp.393–401, Sep. 2006.
14) J.So and N.Vaidya, "Multi-channel MAC for ad hoc networks: Handling multi-channel hidden terminals using a single transceiver," Proc. 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing, pp.222–233, May 2004.
15) J.Zhang, G.Zhou, C.Huang, S.-H.Son, and J.-A.Stankovic, "TMMAC: An energy efficient multi-channel MAC protocol for ad hoc networks," Proc. IEEE International Conference on Communications, pp.3554-3561, Jun. 2007.
16) G.-J.Yu and C.-Y.Chang, "An efficient cluster-based multi-channel management protocol for wireless ad hoc networks," Computer Communications, vol.30, pp.1742-1753, 2007.
17) V.Kanodia, A.Sabharwal, and E.Knightly, "MOAR: a multi-channle opportunistic auto-rate media access protocol for ad hoc networks," Proc. 2004 Broadband Communications, Networks and Systems, pp.600–610, 2004.
18) F.Chen, H.Zhai, and Y.Fang, "An opportunistic multiradio MAC protocol in multirate wireless ad hoc networks," IEEE Transactions on Wireless Communications, vol.8, pp.2642–2651, 2009.
19) B.Han and K.Kawanishi, "Congestion window limit multi-channel MAC protocol: Employing random channel selection strategy for multi-hop ad hoc networks," Proc. 12th International Symposium on Wireless Personal Multimedia Communications, Sep. 2009.
20) "The network simulator *ns*2," Web site: *http://www.isi.edu/nsnam/ns/*.