Cross-Layer Design for Exposed Node Reduction in Ad Hoc WLANs

Article *in* IEICE Transactions on Communications · January 2018

DOI: 10.1587/transcom.2017CQP0015

Emilia Weyulu
Max Planck Institute for Informatics

Masaki Hanada
Tokyo University of Information Sciences

Hidehiro Kanemitsu
Tokyo University of Technology

Eun-Chan Park
Dongguk University

IEICE
TRANSACTIONS
on Communications

VOL. E101-B NO. 7
JULY 2018

A PUBLICATION OF THE COMMUNICATIONS SOCIETY
The Institute of Electronics, Information and Communication Engineers
Kikai-Shinko-Kaikan Bldg., 5-8, Shibakoen 3chome, Minato-ku, TOKYO, 105-0011 JAPAN

PAPER *Special Section on Communication Quality in Wireless Networks*

Cross-Layer Design for Exposed Node Reduction in Ad Hoc WLANs

Emilia WEYULU[†], ***Nonmember***, **Masaki HANADA**[†a)], **Hidehiro KANEMITSU**[††], ***Members***, **Eun-Chan PARK**[†††], ***Nonmember***, ***and*** **Moo Wan KIM**[†], ***Member***

SUMMARY Interference in ad hoc WLANs is a common occurrence as there is no centralized access point controlling device access to the wireless channel. IEEE 802.11 WLANs use carrier sense multiple access with collision avoidance (CSMA/CA) which initiates the Request to Send/Clear to Send (RTS/CTS) handshaking mechanism to solve the hidden node problem. While it solves the hidden node problem, RTS/CTS triggers the exposed node problem. In this paper, we present an evaluation of a method for reducing exposed nodes in 802.11 ad hoc WLANs. Using asymmetric transmission ranges for RTS and CTS frames, a cross-layer design is implemented between Layer 2 and 3 of the OSI model. Information obtained by the AODV routing protocol is utilized in adjusting the RTS transmission range at the MAC Layer. The proposed method is evaluated with the NS-2 simulator and we observe significant throughput improvement, and confirm the effectiveness of the proposed method. Especially when the mobile nodes are randomly distributed, the throughput gain of the Asymmetric RTS/CTS method is up to 30% over the Standard RTS/CTS method.
key words: *RTS/CTS, cross-layer, exposed node, AODV, NS-2*

1. Introduction

Recent advances in wireless communications design have greatly reduced the operational costs and energy requirements of wireless network equipment. This has popularised ad hoc wireless networks that do not rely on any fixed infrastructure, with applications in social and commercial enterprises. In these self-organizing, self-configuring networks, nodes build automatic connections to other nodes; and they each attempt to access the shared wireless medium on their own. The lack of centralized infrastructure to coordinate the activities of nodes gives ad hoc networks the advantage of simplicity but also makes them prone to interferences [1].

Among multiple nodes that compete for channel access, if only one node makes a transmission attempt, the packet is delivered successfully. However, if multiple nodes attempt to transmit simultaneously, collisions may occur. The commonly used IEEE 802.11 wireless networking standard defines medium access control (MAC) specifications for channel contention resolution [2]. The IEEE 802.11 Distributed Coordination Function (DCF) implements CSMA/CA as a channel access control method, and to reduce collisions. The clear channel assessment (CCA) in CSMA/CA uses two techniques to combat interference: mandatory physical carrier sense that monitors the signal strength of the channel, and optional virtual carrier sense that uses the Request-To-Send/Clear-To-Send (RTS/CTS) handshake to reserve the wireless medium prior to transmission.

Physical carrier sensing allows a node to initiate a transmission only if all the other nodes in its sensing range are idle. This helps to avoid collisions effectively as long as the potential interfering nodes are able to sense the radio signal from the source node [3]. However; if a node happens to be outside the reception range of the source node, and starts transmitting at the same time, it is bound to cause collisions at the destination node. This is known as the hidden node problem.

Virtual carrier sensing was introduced as an optional mechanism to solve the hidden node problem. It uses a four-way handshake as a way to reserve the shared wireless medium [4]. The handshake method works by exchanging 'Request to Send' and 'Clear to Send' (RTS/CTS) messages before the actual communication takes place. In Fig. 1, when node C wants to transmit data to node B, it first sends an RTS message to destination node B. If the RTS message is successfully received by node B without suffering collisions, node B will respond with a CTS packet to node C. After this, node C can start sending the data to node B. When the transmission of the data is complete, node B sends an acknowledgement (ACK) to node C to confirm correct re-

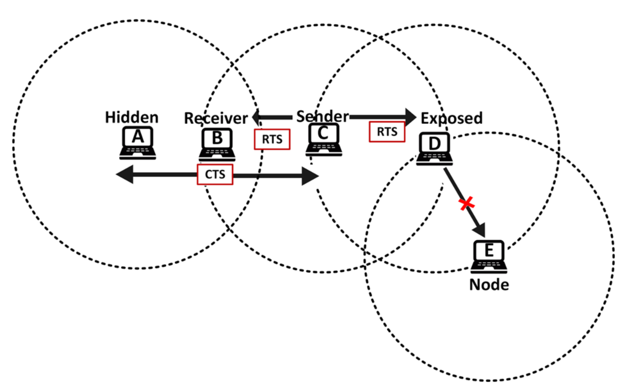


Fig. 1 Hidden and exposed nodes in a wireless network.

Manuscript received August 17, 2017.
Manuscript revised November 30, 2017.
Manuscript publicized January 22, 2018.
[†]The authors are with Department of Information Systems, Tokyo University of Information Sciences, Chiba-shi, 265-8501 Japan.
[††]The author is with School of Computer Science, Tokyo University of Technology, Hachioji-shi, 192-0982 Japan.
[†††]The author is with Department of Information and Communication Engineering, Dongguk University, Seoul, Republic of Korea.
a) E-mail: mhanada@rsch.tuis.ac.jp (Corresponding author)
DOI: 10.1587/transcom.2017CQP0015

ception of the data.

Other nodes in the network monitor for RTS/CTS frames and defer their transmissions for the time duration specified within the RTS/CTS messages. This time duration is called the Network Allocation Vector (NAV) and specifies the period of time for which the channel will remain busy. When they have a packet to send; deferring nodes allow sufficient time to elapse to ensure that the current RTS/CTS and data/ACK sequence completes before attempting a transmission. Although RTS/CTS reduces collisions from hidden nodes, it causes another problem known as the exposed node problem. Exposed nodes are prevented from communicating with other nodes in their transmission ranges because they are close to the source node and overhear the RTS frame. In Fig. 1, the transmission from node D to node E will not be allowed because node D overhears the RTS frame from node C and defers its transmission. The back-off mechanism of exposed nodes in ad hoc networks contributes to the underutilization of network capacity.

In this paper, we present a method for reducing the number of exposed nodes by improving the standard RTS/CTS handshake method. The proposed method builds on the Asymmetric RTS/CTS method introduced in [5] which uses different transmission ranges for RTS and CTS frames to reduce the number of exposed nodes in an ad hoc network. The method in this paper goes a step further by implementing a cross-layer design between layers 2 and 3 (i.e. MAC layer and the routing layer) of the Open Systems Interconnection (OSI) model [6]. With the cross-layer design, we are able to control the transmission range of subsequent RTS frames using routing information from the selected best-route i.e. the received signal strength (RSSI) of the next-hop node.

Additionally, we investigate the relationship between the transmitted power of RTS packets and carrier sense, and how the power with which an RTS packet is transmitted can be optimized to increase spatial reuse and improve network performance. By adopting a cross-layer design between the MAC layer and the routing layer, we are able to select the most optimal route to the destination node with a reduced number of exposed nodes, while still using the lowest hop count. Through simulations, we observe that the improved Asymmetric RTS/CTS method shows significant throughput gain over the standard RTS/CTS method.

The rest of this paper is organized as follows. In Sect. 2, we give an overview of related research work on the exposed node problem and cross-layer designs, and review the drawbacks of their proposed solutions. We focus on the Asymmetric RTS/CTS idea and review some of its current shortcomings and how it can be improved upon. In Sect. 3, we introduce the proposed cross-layer design used to improve the performance of the Asymmetric RTS/CTS method. We show how a neighbour node's RSSI can be used to estimate the RTS transmission range of subsequent data packets. The simulation setup and evaluation of the proposed method is described in Sect. 4 and a comparison (in terms of overall network throughput) with the standard RTS/CTS method and an existing method is presented. In Sect. 5, an overview of the assumptions considered in the proposed method is presented. Concluding remarks are in Sect. 6.

2. Related Work

2.1 Exposed Nodes in 802.11 Networks

The suggestion that exposed nodes should be able to transmit to other nodes in their transmission ranges without harming an ongoing transmission has been made several times in literature [3], [7], [8]. Although the IEEE 802.11 RTS/CTS mechanism partially solves the exposed node problem, this is only in cases where the nodes in the network are synchronized and packet sizes and data rates are the same for both transmitting nodes. In such scenarios, if a node hears an RTS from a neighbouring node but cannot hear a subsequent CTS within a certain amount of time, it can deduce that it is an exposed node and it is permitted to transmit to its neighbouring nodes.

A proposal where nodes identified themselves as exposed nodes and transmitted opportunistically was proposed in [7]. The method permitted a secondary DATA transmission to occur in parallel with the primary DATA transmission, without invoking the RTS/CTS mechanism for the secondary DATA transmission. The method in [7] improved network throughput but its application was limited to cases where the secondary transmission could fully align itself with the primary transmission. Otherwise the algorithm reverted back to the standard 802.11 RTS/CTS method. Synchronizing node's communications in ad hoc networks is a complicated task because different node's local clocks may shift over time, and at different rates, causing the perceived time and period of each node to be different [9]. This issue is further exacerbated in the case of mobile ad hoc networks (MANETs).

In [3],the authors proposed an improved channel access mechanism using frame aggregation and augmented CSMA. This was to allow transmission from exposed nodes outside the carrier sense range of the destination and source nodes. The method improved throughput by reducing virtual carrier sensing signalling overhead. Using the transmit power levels of the control frames, the authors obtained better throughput than existing methodology. The power control method used in [3] shows that adaptively controlling the transmit power of control frames allows exposed nodes to reuse spatial slots and helps to improve the throughput performance of the network.

A similar strategy was used in [8] where the researchers used power control to tune the physical carrier sense transmit power. The authors tuned the transmit power so as to have the carrier sense range almost equal to the interference range. In this way, the hidden nodes which were within the interference range, but outside the carrier sense range of the source node, could be eliminated. However; tuning the carrier sense range to be the same as the interference range fails to eliminate the hidden nodes that are within the range of the destination node and can lead to more collisions at the destination node.

Another approach that attempts to solve both problems is the Dual Busy Tone Multiple Access (DBTMA) method proposed in [10]. DBTMA essentially solves the hidden and the exposed node problems by using two out-of-band busy tones to protect the RTS packets and the DATA packets from interfering nodes. This protocol however assumes separate channels or data streams for tones and data. Although it is technically possible for wireless devices to communicate using multiple channels simultaneously, the MAC protocol in 802.11 networks is designed for a single channel only [11]. Hence, we only consider single channel scenarios in our research.

Other methods proposed in literature to solve the exposed node problem are such as that described in [12] that temporarily disables the carrier sense mechanism. The method in [12] called Selective Disregard of NAVs (SDN) selectively ignores certain physical carrier sense and NAVs. This method however needs additional functionalities to be implemented in the nodes and lacks compatibility with the IEEE standard. Other MAC protocols based on Multiple Access with Collision Avoidance (MACA) were proposed in [13] that exploit control gaps between the RTS/CTS exchange and the subsequent DATA/ACK. The method used in our paper uses existing IEEE 802.11 MAC protocol techniques and does not require the introduction of new control frames.

The method in [14] modifies the transmission power of RTS/CTS frames and DATA packets to solve both the hidden and exposed node problems. The authors in [14] proposed a modification of the CTS transmission range by increasing the power at which a CTS message is sent. Similar to our approach, this method used information from the next-hop neighbour to adjust CTS transmission range. This method was limited to scenarios where a 'balanced' range of CTS could be found and performance was proportional to both the network size and the number of additional nodes. A significant drawback of this method is the requirement of global knowledge of transmission powers and node locations, which is often not possible in real environments.

In [15], the authors proposed an interesting approach that considers transmission rates of MAC control frames and DATA packets to solve the hidden node problem. The method called Power control MAC (PMAC), is enhanced using a power control scheme that is used jointly with the adjusted transmission rates to solve the exposed node problem. PMAC uses power control for DATA and ACK packets while RTS and CTS packets are still sent with a higher transmit power in order to block potential interferers. However, this means that nodes that receive the RTS packets will still set their NAV timer and back-off from accessing the channel, hence the exposed node problem persists. Throughput performance of our proposed method is compared with that of PMAC in Sect. 4.3.

Amongst all the reviewed methods, it is difficult to find a method that reduces both hidden and exposed nodes while adhering to the IEEE 802.11 MAC protocol standard. Generally, existing research shows that solving both the hidden and exposed node problems seems to be an almost impossible task as these are two complementary problems [3], [8]. There is usually a trade-off between the hidden node problem and the exposed node problem.

2.2 Cross-Layer Designs in Wireless Networks

To satisfy performance requirements of modern wireless networks, it is important to consider cross-layer communication in our effort to regulate access to the shared wireless channel. Cross-layer designs help to break down the traditional waterfall concept of the OSI model and provide inter layer communication between non-adjacent layers. This allows information sharing through the layer boundaries, increasing network performance and reliability. Cross-layer designs are being increasingly studied as an approach to channel allocation and spatial reuse [1]. In this section, we review some existing cross-layer designs that attempt to improve wireless throughput and address the differences between these approaches and the method used in this paper.

In [16], the authors proposed a cross-layer design to improve spatial reuse. The proposed method addressed the problem of on-demand dynamic channel selection in CDMA-based ad hoc networks by jointly considering MAC layer scheduling and physical layer requests. Although the method provides contention free transmissions and helps to eliminate the dependency of the number of channels on network size, it was optimized using end-to-end metrics. Additionally, this method is limited to single-hop networks because it did not consider factors such as routing and the location of nodes.

In [17], the authors proposed a method that used a cross-layer design for wireless channel assignment and reservation. Similar to our proposal, the cross-layer design was implemented between the routing layer and the MAC layer. However, this method used a multi-channel approach by first using the number of available channels as a routing index and then assigned a channel for data transmission. Although multi-channel approaches present significant throughput gains over single channel approaches, we limit comparison of our proposal to single-channel approaches at this time. The associated high costs of multiple transceivers and lack of coordination between clients that use different channels are issues hindering multi-channel adoption in wireless networks.

2.3 Asymmetric RTS/CTS

The research reported in [5] provides a first glance and theoretical model for reducing exposed nodes in IEEE 802.11 ad hoc WLANs using the concept of Asymmetric RTS/CTS. Similar to the PMAC method described in [15], the authors in [5] also considered the effect that transmission rate has on transmission range. Transmission range is the distance within which a transmitted message can be correctly received interference free. In IEEE 802.11 networks, transmission rate determines transmission range and the transmission rate can be tuned to optimize wireless network performance

Table 1 Transmission rate vs distance.

Transmission rate (Mbps)	802.11a Indoor (m)	802.11a outdoor (m)	802.11b/g Indoor (m)	802.11b/g outdoor (m)
1	_	_	124	610
6	50	304	91	396
11	_	_	48	304
18	33	183	54	183
54	13	30	27	76

[15]. Lower transmission rates can be demodulated across greater distances than higher transmission rates. This means that decreasing the transmission rate increases the effective range of a wireless node and vice versa. Table 1 shows the relationship between transmission rate and distance for Cisco Aironet CB21AG and PI21AG Wireless LAN Client Adapters [18].

However, tuning the transmission rate of a wireless node also affects the network sensing and interference ranges. This in turn affects the number of nodes that withhold their transmissions to avoid interfering with an ongoing transmission i.e hidden nodes and exposed nodes [19].

To avoid the increase of hidden nodes while solving the underutilization of the network capacity by reducing exposed nodes, the authors in [5] tuned the effective range of RTS by adjusting the RTS transmission rate to the maximum. This reduced the RTS transmission range and made it possible for RTS frames to reach the next-hop node while excluding surrounding nodes from its carrier sensing range. Reducing the transmission range of RTS frames means some of the exposed nodes in the original RTS transmission range are eliminated. The transmission rate of CTS was set to a lower rate so that CTS frames can reach all possible nodes that would cause collisions at the destination node. Based on this strategy, the transmission ranges for the RTS and CTS frames are thus asymmetric.

RTS frames do not need to have such a wide transmission range, as they only need to reach the destination node (or next-hop node) in order to provoke a CTS response [5]. Thus if the transmission range of RTS is set to the minimum distance, only reaching the destination node, this is enough to provoke CTS from the destination node. By having the RTS transmission range large enough to only reach the next-hop node, the total number of exposed nodes in the network can be reduced or even be eliminated in cases where the RTS range is completely included in the CTS range.

Regarding other control frames used in CSMA such as the ACK packet, they only need to be received by the source node and are hence also transmitted using the adjusted RTS transmission range. CTS frames on the other hand, need to have a large transmission range so that the data frame reception at the destination node is protected.

2.4 Issues with Asymmetric RTS/CTS

In [5], the researchers focused primarily on evaluating the performance of the basic Asymmetric RTS/CTS idea and on validating the effects of the MAC layer improvements. They thus limited their investigations to one-hop node communications. The reduced RTS transmission range was evaluated to be effective with regards to communication with neighbours at a one-hop distance and not beyond. This was possible because the wireless scenario for the evaluation had nodes located at known, arbitrary distances before the RTS transmission range was set. Transmissions outside of this range was not possible as the RTS transmission range had to be reset manually. In addition, resetting the RTS transmission range to cater for two-hop nodes reverted the idea back to the standard RTS/CTS mechanism and this resulted in the failure of the proposed method.

In real-life networks, due to the limited transmission range, routes between nodes are normally created through several hops. We thus need to define a procedure that optimizes Asymmetric RTS/CTS for multi-hop communication. Traditional end-to-end communication uses routing protocols that efficiently select an appropriate node among its neighbouring nodes as the next-hop node in order to reach the destination node. Furthermore, multiple forwards are common for multi-hop communications and may force a significant change of the carrier sensing threshold from when only one-hop flows are considered. To optimize the end-to-end performance of multi-hop flows, carrier sensing range and spatial reuse as well as next-hop distance must be appropriately addressed.

In this paper, we make improvements to the Asymmetric RTS/CTS idea in order to accommodate end-to-end, multi-hop communication. This is made possible by implementing a cross-layer design that addresses both MAC layer and routing issues. By improving the Asymmetric RTS/CTS idea to accommodate end-to-end communication, we further investigate the impact that different node topologies has on the performance of the proposed idea.

The authors in [5] experimented with multi-rate transmission of RTS and CTS frames in order to control transmission range. In this paper, we directly adjust the transmission ranges of the control frames without considering transmission rate. This is done for simplicity and to avoid complications with the PHY layer convergence procedure (PLCP) preamble whose transmission rate cannot be changed. We adjust the RTS transmission range based on the location information of the next-hop node obtained through a layer 3 routing protocol.

3. Basic Idea

3.1 The AODV Routing Protocol

To sufficiently evaluate the effectiveness of the Asymmetric RTS/CTS idea, we consider end-to-end node communication and use a cross-layer design with MAC and routing protocols. Using the cross-layer design between layer 2 and layer 3, the wireless nodes in our ad hoc network must cooperate in routing the data packets from the source node to the destination node. One of the most popular routing protocols in ad hoc networks is the Ad Hoc On-Demand Distance Vector

Table 2 AODV RREQ.

RREQ format	
`Source address`	IP address of source node
`Source seq. number`	To maintain freshness info about the route to the source
`Destination address`	IP address of destination node
`Dest. seq. number`	Specifies how fresh a route to the destination must be before it is accepted by the source
`Hop count`	Specifies the number of hops between the source and the destination

(AODV) routing protocol. AODV uses a form of reactive routing that establishes routes to the destination nodes only when they are requested by the source nodes [20].

AODV has two important functions that give it an advantage over other routing protocols, route establishment by initiating a route discovery process and maintaining the active routes. This means finding alternative routes in cases of link failures and deleting routes when they are no longer needed. Each node in the network maintains a routing table; with routing information entries to its neighbouring nodes, and two separate counters: a node sequence number and a broadcast-id. When a source node wants to communicate with a destination node, it increments its broadcast-id and initiates path discovery by broadcasting a route request (RREQ) message to its neighbours. The RREQ contains the fields given in Table 2.

Intermediate nodes between the source node and the destination node first check whether the RREQ is new, whether it is meant for them or whether they have a direct route to the destination node; otherwise they rebroadcast the RREQ. When a node with a direct route to the destination node receives the RREQ, it unicasts a Route Reply (RREP) message back to the source node. AODV's routing algorithm gives the network the flexibility to allow nodes to enter and leave the network at will [20].

AODV uses hop count for choosing the best route to transfer data from a source node to a destination node; and nodes that need to send data broadcast a request for connection. The intermediate nodes forward the message and take note of the node that requested the connection. Thus, they create a series of temporary routes back to the requesting node. In AODV, nodes learn of their neighbours in one of two ways; whenever a node receives a broadcast packet from a neighbour, it updates its local connectivity information (i.e. routing table) to ensure that it includes this neighbour [20]. The second way is to obtain neighbourhood information is through the use of HELLO messages. Every HELLO_INTERVAL, measured in seconds, a node that has not sent a broadcast (e.g. an RREQ or an appropriate layer 2 message) within the last HELLO_INTERVAL, generates a broadcast RREP with TTL = 1 second to its neighbours. When a node receives a HELLO message from a neighbour node, it updates its route information associated with that neighbour node in its routing table.

To adopt end-to-end multi-hop communication into the Asymmetric RTS/CTS idea, we need an efficient method to transmit the Data packet† from the source node to the destination node, that takes into account the reduced transmission ranges for RTS frames. This we achieve by modifying the transmission range for RTS packets i.e how far an RTS packet will be transmitted, based on the next-hop node selected by the AODV protocol. The 'new' RTS transmission range is then used to forward the subsequent Data packet from the source node to the destination node after the route has been established. All nodes taking part in the communication process between the source node and the destination node adopt the same procedure until the Data packet has been successfully delivered to the destination node.

3.2 Adaptive RTS Transmission Range

In general, RSSI measurements are used to obtain the position information of wireless nodes in WLAN environments. In 802.11, RSSI is an indication of the power present in a received radio signal and can either be a negative, or a positive value [21]. It is often expressed in decibels (db), or as percentage value between 1-100, and the closer the figure is to zero, the better. We use the RSSI from the next-hop (i.e $RSSI_{Ni}$), from the best route selected by the AODV routing protocol to estimate the transmit power (i.e P_t) that a packet will need to be transmitted with to be received by that next-hop node. In other words, we infer P_t from the RSSI to determine when a transmitted signal will be within a certain node's transmission range [9].

As mentioned in Sect. 2, we use the RSSI of AODV routing messages sent and received during the path discovery process to estimate and set the transmit power of RTS packets (i.e RTS_{PT}) for subsequent Data packet transmissions. In this way, we adjust the RTS transmission range to be just adequate enough to reach the next-hop node. Similar to [5], RTS packets only need to reach the next-hop node in order to elicit a CTS response from that node. By setting RTS_{PT} to be equivalent to $RSSI_{Ni}$, and limiting the RTS transmission range to only reach the next-hop node, we exclude exposed nodes that would have been previously included when using the IEEE 802.11 Standard RTS/CTS RTS transmission range.

We do not adjust the transmit power of CTS packets and therefore its transmission range remains unaltered. In this way, our approach is similar to the Asymmetric RTS/CTS method described in [5]. However, using $RSSI_{Ni}$, we improve the Asymmetric RTS/CTS method by dynamically tuning the RTS transmission range for the Data packet path from the source node to the destination node.

Below is a detailed description of how the RTS transmission range for DATA packets is computed. Suppose the transmission is from Source node i to Destination node j. For this description, we assume there is no existing route between Source i and Destination j.

†Data Packet in our paper refers to the packet of information, different from the control packets used to discover and maintain routes i.e RREQ, RREP, RRER, HELLO

1. Source i broadcasts an RREQ packet to its neighbours.
2. Intermediate nodes between Source i and Destination j receive the RREQ and verify that it is a new RREQ by checking its sequence number. If the RREQ is verified to be new, the intermediate nodes record its RSSI value and rebroadcast it.
3. The RREQ is received by Node_$j-1$ which has a direct route to Destination j, Node_$j-1$ also records the RREQ's RSSI value and unicasts an RREP back to Source i.
4. Intermediate nodes again record the RSSI from the RREP packet and propagate the RREP towards Source i using cached reverse route entries
5. When the RREP makes it back to Source i, Source i records the RSSI value of the RREP packet. Source i uses the recorded RSSI value from the RREP packet to estimate the transmit power of RTS packets (i.e RTS_{PT}) needed to reach the neighbour node when sending the DATA packet to Destination j.
6. Intermediate nodes receiving the DATA packet from Source i for forwarding to Destination j use the same strategy employed in Step 5.

In cases where routes are discovered through the use of HELLO messages, RSSI values are recorded from the HELLO messages at that moment.

We make note of the fact that information that can be obtained from AODV is only the distance between a node and its next-hop neighbour nodes. With this information, each node cannot determine whether it is an exposed node or not, i.e each node cannot determine whether it should defer its transmission or it should transmit. There are two reasons for this, firstly, the exposed node problem is not only related to the position of source nodes but also to the position of destination nodes. Secondly, each node may estimate the distance to its neighbour node (by using a routing protocol i.e. AODV) but it cannot know the location conditions of nodes close to the neighbour node that are not within its transmission range. Therefore, the exposed node problem cannot be effectively resolved by only using information about neighbour nodes received from AODV.

Figure 2 explains the recording of the next-hop's RSSI whenever HELLO messages are exchanged. Every node that participates in the communication between the source node and the destination node follows this procedure.

After recording the RSSI value from the HELLO packets, a node receiving a Data packet for transmission first checks its routing table to make sure it has a route to the requested destination. If there is no route present, the route discovery process according to the AODV routing protocol is initiated. Nodes inform the MAC layer to adjust the transmit power of RTS packets i.e RTS_{PT} to the RSSI of the next-hop node i.e $RSSI_{Ni}$ based on the results of the route discovery and forward path set-up process. Figure 3 shows an example of a random topology where Node 3 sets its RTS transmission range to 35m (based on received RSSI = 1.55e-07) for Destination Node 5 and 11m (based on received RSSI = 1.51e-06) for Destination Node 4 (next-hop Node 2). Figure 3 also shows the adjusted AODV routing table that includes the recorded RSSI value.

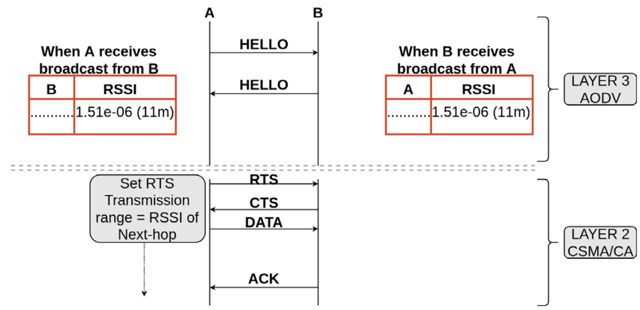
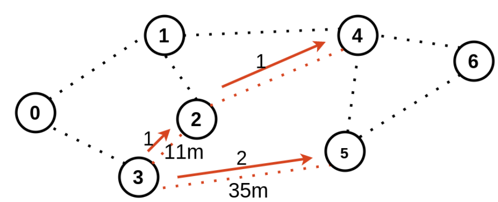


Fig. 2 Recording RSSI from broadcast messages.



Routing table of Node 3

IP	Seq. no	Destination	Hop count	Next hop	RTS Range
10.0.1.20	1	5	1	5	35m (RSSI = 1.55e-07)
10.0.1.20	2	4	2	2	11m (RSSI = 1.51e-06)

Fig. 3 Node 3 routing table showing how RTS transmission range will be adjusted.

Algorithm 1 shows pseudocode for the algorithm used to record $RSSI_{Ni}$ in an AODV routing table and the calling of the Layer 2 function that adjusts the RTS transmission range. We only use the RREQ and RREP messages here as examples for simplicity. Algorithm 2 shows pseudocode for the Layer 2 function that adjusts the RTS transmission range based on $RSSI_{Ni}$.

Although our proposed method is evaluated using the AODV protocol, we believe it can be extended to other ad-hoc routing protocols. The advantage of AODV is that it is adaptable to highly dynamic networks. However, nodes may experience large delays during route construction, and link failure may initiate another route discovery, which introduces extra delays and consumes more bandwidth as the size of the network increases.

4. Simulation and Results Analysis

4.1 Simulation Environment

The proposed changes to the Asymmetric RTS/CTS idea were implemented using the Network Simulator-2 (NS-2), and simulation scenarios for various topologies carried out. NS-2 is an event driven, object oriented network simulator enabling the simulation of a variety of local and wide area networks, and is widely used in communication networks research [22].

Algorithm 1

```
Obtain RSSI from AODV routing control packets

if Packet == "RREQ" then
    if Packet is not in RREQ list then
        add entry to Routing Table
        record RSSI
        add Packet_ID into RREQ list
    else if Packet_Dest == Current_Node_Address then
        Prepare RREP and send it
    else
        Forward Packet
    end if
else
    Ignore Packet
end if

if Packet == "RREP" then
    if Packet is not in RREP list then
        add entry to Routing Table
        record RSSI
        add Packet_ID into RREP list
    else if Packet_Dest == Current_Node_Address then
        Prepare RREP and send it
    else
        Forward Packet
    end if
else
    Ignore Packet
end if

if Packet == "DATA" then
    if Packet_Dest == Current_Node_Address then
        Packet reached destination
    else
        Check routing table for RSSI_Ni
        Call Layer 2 function: Set_RTS_Range
        Send Data Packet to next-hop
    end if
end if
```

Algorithm 2

```
Set RTS Transmit Power

Set_RTS_Range
Set MaxHops
Hops = 1
Input: RSSI_Ni

while Hops < MaxHops do
    RTS_PT = RSSI_Ni + margin
    where i in {1, 2, .., MaxHops}
    Send RTS packet
    Break from While
    Hops = Hops + 1
end while
```

Our proposed method assumes wireless networks that are primarily used for "Groupware" or "mobile collaborative applications" [23]. These are ad hoc networks where users engage in collaborative tasks while on the move within a certain area. Groupware gives users capabilities to broadcast over secure networks, enabling multi-party conferencing in real time. Such scenarios can be found at workshops, conference meetings or industry exhibitions. With the current expansion of mobile networking services, future wireless networks need to be conducive for collaborative, self-configuring multi-hop ad hoc networks.

The above-mentioned collaboration scenarios usually need to support voice and video traffic, hence WLAN infrastructure must provide sufficient bandwidth capacity to ensure endpoints can successfully make high-quality voice and video calls. In particular the number of simultaneous voice or video bidirectional traffic streams in the WLAN channel is a critical capacity consideration. Networks offering real-time traffic such as voice or video are often pushed beyond their capacity which leads to performance slowing for all connected parties. Hence it is important to evaluate the proposed method in case of a fully saturated network where the number of flows is identical to the number of nodes in the network, which is referred to as "full-capacity flows" hereafter.

Fully saturated network scenarios where each station always has a packet for transmission have been used in widely cited literature such as [24] and most recently in [25] to investigate WLAN performance. For these reasons, we evaluate our proposed method not only in the case of low-traffic scenarios (i.e 3 traffic flows) but also in the case of fully saturated traffic scenarios (i.e full capacity flows).

Examples of the simulated grid and random topologies in NS-2 are shown in Figs. 4 and 5 depicting 9 nodes.

For grid topologies, the distance between nodes is set to 70 m and we assume an environment where each node can communicate with each of its adjacent nodes that are within a transmission range of 70 m. To simulate the Standard RTS/CTS method in grid topologies, we set both transmission ranges for RTS frames and CTS frames to 140 m. For the Asymmetric RTS/CTS method, we set transmission range for RTS frames to be equivalent to the distance of the next-hop node, i.e. 70 m and CTS frames to 140m. We simulated grid topologies of 9 nodes (3x3), 16 nodes (4x4), 25 nodes (5x5) and 100 nodes (10x10). The Asymmetric RTS/CTS method and the Standard RTS/CTS method will from hereafter be referred to as A-RTSCTS and S-RTSCTS respectively, in the simulations results, figures and discussions.

In random topologies, nodes were positioned following a uniform random arrangement. For simplicity and to keep consistency, the transmission ranges for both the RTS and CTS frames were set to 140 m for S-RTSCTS, while for A-RTSCTS, the RTS transmission range was dynamically determined based on $RSSI_{Ni}$ as described in Sect. 3. The transmission range for CTS frames was again set to 140 m. Similar to grid topologies, we again simulated topologies of 9, 16, 25 and 100 nodes. For consistency, the topology area sizes that were used for the respective grid topologies were also used in the case of random topologies. The topology areas increased by a factor of 1.4 for each increase in topology size, starting at 180x180 for 9 nodes.

Traffic flow in both topologies consisted of Constant Bit Rate (CBR) flows of packet size 500 bytes. CBR data provides low-latency traffic with predictable delivery char-

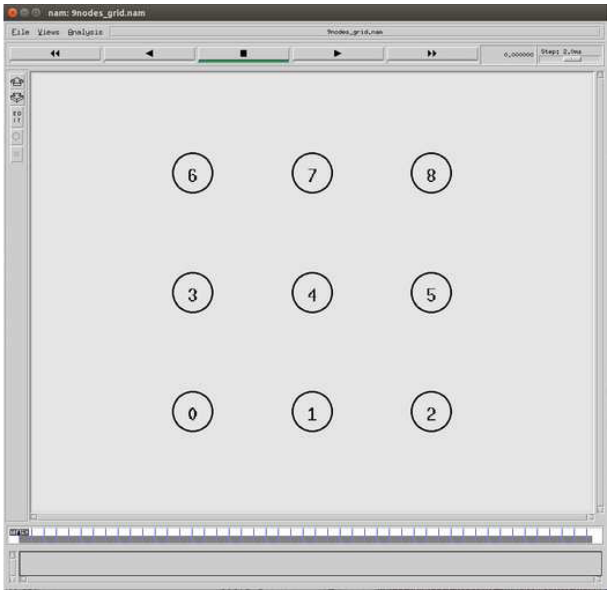


Fig. 4 Grid topology of 9 nodes in NS-2.

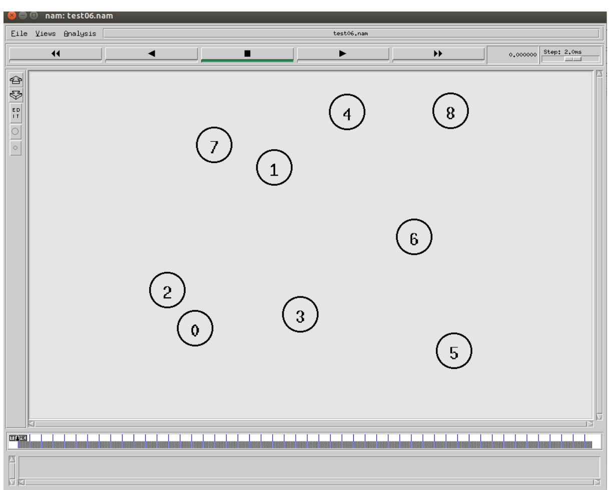


Fig. 5 Random topology of 9 nodes in NS-2.

acteristics and is convenient for evaluating network performance [26]. There were 4 different traffic flows simulated i.e 3, 6, 9 and full capacity flows†. This was to evaluate how A-RTSCTS performs when network capacity is varied. The packet interval, in seconds, was set to 0.1, 0.05 and 0.01. The packet interval is the time interval between two CBR packet generations. In all topologies, nodes started their transmissions at 1.0 seconds and stopped at 60 seconds for a duration of 59 seconds. In all instances, the performance of the A-RTSCTS method was compared to the performance of the S-RTSCTS method.

To simulate mobility in both grid and random topologies, the node speed was set to 2 metres per second (i.e walking speed). This setting assumed that the nodes move slowly enough such that $RSSI_{Ni}$ recorded during the path

†full capacity flows = total possible flows in each network

Table 3 Simulation parameters and conditions.

Frame Type	Transmission range: Standard RTS/CTS	Transmission range: Asymmetric RTS/CTS
RTS	140m	Next-hop node distance
CTS	140m	140m
Other parameters		
Bandwidth	11Mb	
Packet Type	CBR	
Packet Interval	0.1, 0.05, 0.02	
Data packet size	500 bytes	
Propagation model	TwoRayGround	
Routing protocol	AODV	
Simulation conditions		
Topology sizes	9, 16, 25, 100	
No. of traffic flows	3, 6, 9, Full capacity	
Simulation time	60 seconds	
Simulation freq.	x30	

discovery process can still be used as a reliable setting for the RTS transmission range to transfer the Data Packet. In grid topologies, we assumed the nodes move in a grid-like fashion to represent, for example, users moving from one booth to another in an exhibition hall that has grid like boundaries [23]. In random topologies, the Random Waypoint Mobility Model [27] was used.

Table 3 presents the rest of the simulation parameters and conditions used. From the determined AODV routing path, $RSSI_{Ni}$ was recorded and the value inserted into the AODV routing table using the NS-2 program with some source code modifications. To calculate the RSSI, we used the Two-ray Ground Model in order to consider both the direct path and the ground reflection path. When a node had packets to send, it simply searched its routing table for the updated RSSI information and then set its RTS transmission range as previously described in Sect. 3.

In real networks, the interference range of a node may be much larger than its transmission range. Additionally, concurrent transmissions in the same area may cause interferences even when the nodes are two hops away from each other. To accommodate this effect in our simulations, we used an additional RSSI threshold setting to make sure other nodes in the vicinity rejected incoming RTS frames not meant for them. At the MAC level of NS-2, we set the RSSI threshold that restricted how far an RTS frame is allowed to go by ensuring that nodes two-hops away drop incoming RTS packets that have a higher value than the defined RSSI threshold. An incoming RTS packet's RSSI is compared to the RSSI threshold during runtime and if the RSSI level was found to be larger than the threshold, the incoming RTS packet is simply discarded and the node does not set NAV.

The nodes that rejected the RTS frames not meant for them were thus no longer exposed nodes and could communicate with other nodes in their transmission ranges. In this way, we were able to confine RTS packets to a certain transmission range i.e to only the next-hop node. Using the above methods, we were able to reduce the number of exposed nodes in the network.

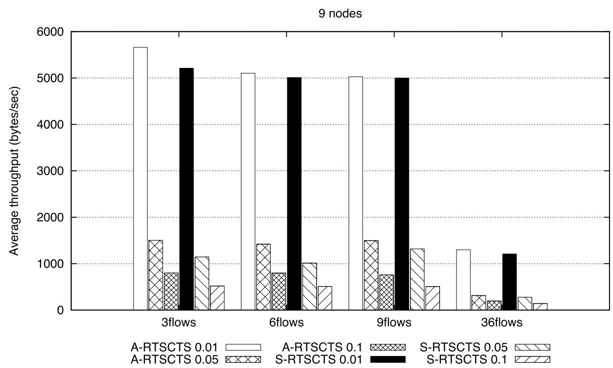


Fig. 6 Traffic generation intervals, 9 nodes.

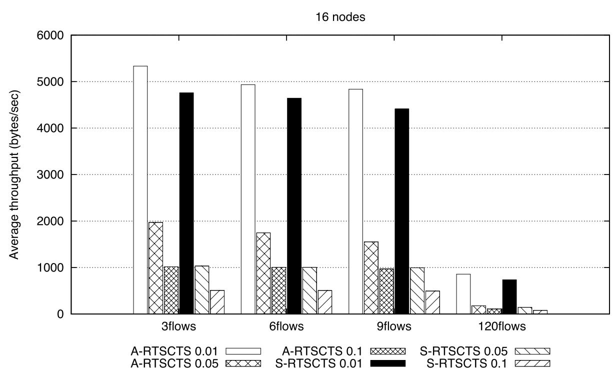


Fig. 7 Traffic generation intervals, 16 nodes.

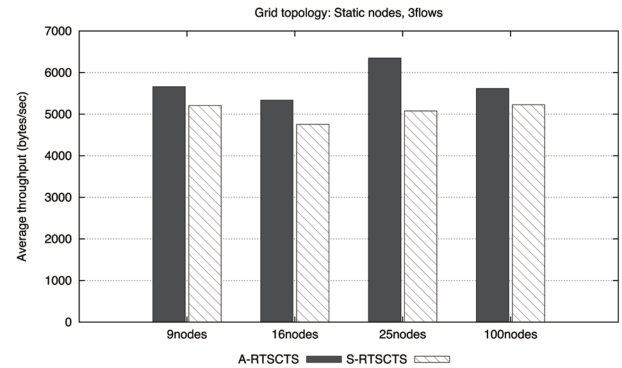


Fig. 8 Average throughput: Static grid topologies, CBR packet interval=0.01, 3 flows.

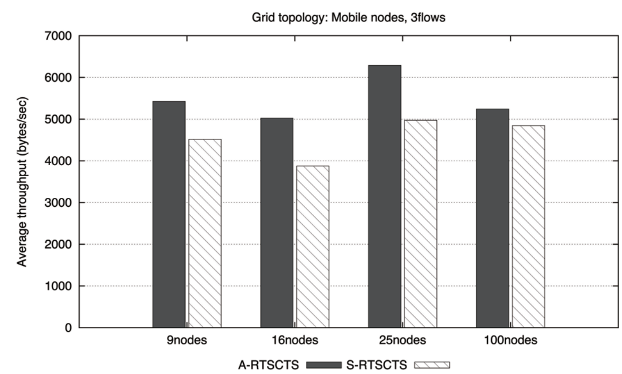


Fig. 9 Average throughput: Mobile grid topologies, CBR packet interval=0.01, 3 flows.

4.2 Comparison with IEEE 802.11 Standard RTS/CTS

We describe the results obtained from simulations comparing the performance of the proposed method, A-RTSCTS, to the standard 802.11 RTS/CTS method, S-RTSCTS. After each simulation, average CBR throughput was calculated and analysed to show the difference in performance between A-RTSCTS and S-RTSCTS. Throughput is the average of successful messages delivered over a communication link, measured as a function of time. We calculate throughput based on the total number of packets received and the packet size divided by the transmission time. Transmission time refers to the time when traffic flow ends minus the time when the traffic flow started as described by Eq. (1). Total network throughput was then calculated based on Eq. (2).

$$Transmission_{time} = T_{end\ time} - T_{start\ time} \tag{1}$$

$$Throughput = \frac{Total\_Recvd_{Packets} \times Packet_{Size}}{Transmission_{time}} \tag{2}$$

Figures 6 and 7 show throughput comparisons based on CBR traffic packet intervals for both A-RTSCTS to S-RTSCTS, for 9 and 16 nodes in static grid topologies. We combined this with differing traffic flows in the network to observe how the proposed method performs in saturated networks operating at different levels of network capacity. Varying the traffic flows in the simulated network allowed for further investigation of which method performed better in a wireless network faced with varying degrees of saturation.

The results in Figs. 6 and 7 were obtained by factoring in the number of flows in each case. From the results, we observe that A-RTSCTS performs well in saturated networks compared to S-RTSCTS which shows no noticeable difference in performance. However, as the packet interval gets larger and the number of traffic flows in the network increase, we see a general trend of decreased throughput in both network topologies. When the topology size in the simulation was varied, care was taken to ensure that the Data Packet size, CBR packet interval rate and the number of flows in the network was similar for each comparison.

Figures 8, 9, 10 and 11 show the average throughput when the network is operating at low network traffic (i.e 3 traffic flows) at CBR packet interval = 0.01 seconds. The results presented are for both static and mobile grid and random topologies, for 9, 16, 25 and 100 nodes. For grid topologies in Figs. 8 and 9, it is difficult to get a clear trend of performance because the number of traffic flows in the network is similar. However A-RTSCTS has a clear performance gain over S-RTSCTS.

In Figs. 10 and 11, the results show better improvement in case of 16 and 25 nodes in comparison to 9 and 100 nodes. In the case of 9 nodes and 100 nodes, we observe less throughput gain for A-RTSCTS over S-RTSCTS. For the 9 node topology, this is because half the nodes are included in the CTS range for both A-RTSCTS and S-RTSCTS so the

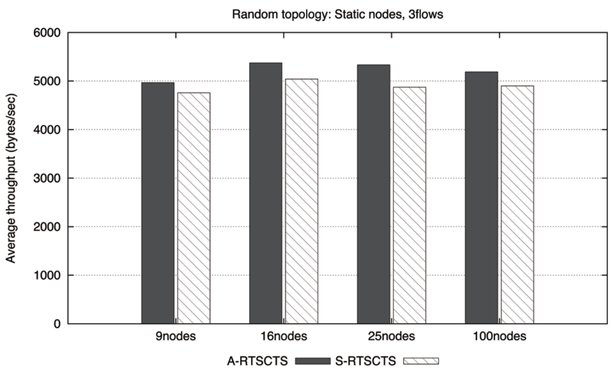


Fig. 10 Average throughput: Static random topologies, CBR packet interval=0.01, 3 flows.

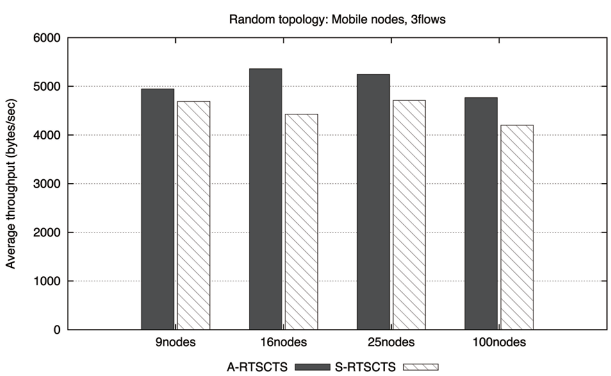


Fig. 11 Average throughput: Mobile random topologies, CBR packet interval=0.01, 3 flows.

difference in throughput is not significant. For 100 nodes, there are less exposed or hidden nodes but because the number of nodes is large, there are many collisions (i.e from signalling) and thus overall throughput is low.

For 16 and 25 nodes, we obtain a much more significant throughput gain between A-RTSCTS and S-RTSCTS because less nodes are exposed or hidden, and can thus transmit to other nodes. From these results, we observe that there is a middle ground in the number of nodes where the proposed method works best. However, we have not yet investigated the optimal number of nodes where our proposal has the most significant gain in throughput and leave this for future work. The improvement ratios obtained using Eq. (3) for the 3 traffic flows in the different topologies show a similar trend to those of full-capacity flows shown in Table 4 and have thus been omitted here.

Figures 12, 13, 14 and 15 show the average throughput when the network is fully saturated (i.e full capacity flows), at CBR packet interval = 0.01 seconds. The results presented are for both static and mobile, grid and random topologies, for 9, 16, 25 and 100 nodes. Each topology scenario simulation was again run for 59 seconds and repeated 30 times to obtain a statistically significant result.

Taking into account the number of traffic flows in the simulated topologies, the simulation results show a general decline in throughput also seen earlier in Figs. 6 and 7, as the number of traffic flows in the network increase. This is

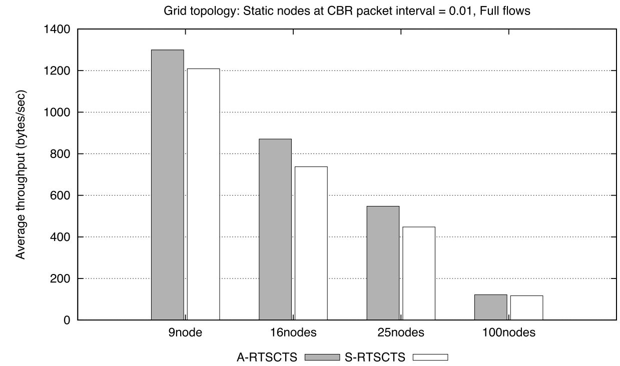


Fig. 12 Average throughput: Static grid topologies, CBR packet interval=0.01, full capacity.

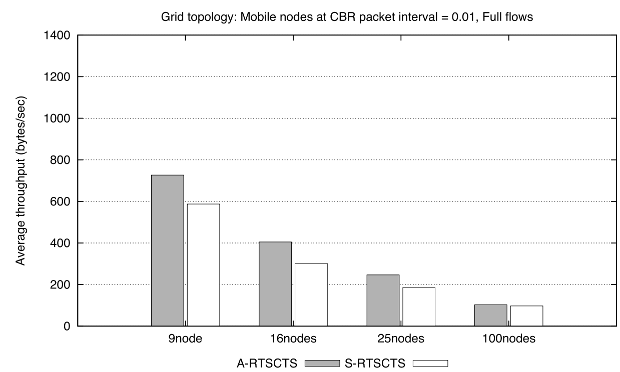


Fig. 13 Average throughput: Mobile grid topologies, CBR packet interval=0.01, full capacity.

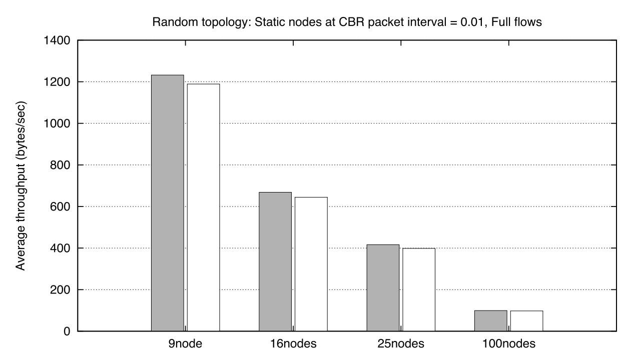


Fig. 14 Average throughput: Static random topologies, CBR packet interval=0.01, full capacity.

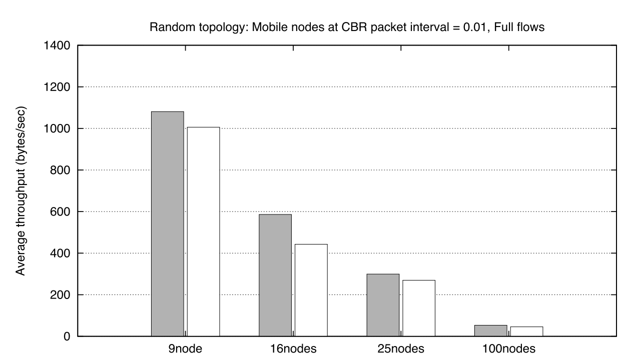


Fig. 15 Average throughput: Mobile random topologies, CBR packet interval=0.01, full capacity.

Table 4 Throughput improvement ratio.

Number of nodes	Static nodes		Mobile nodes	
	Grid topology	Random topology	Grid topology	Random topology
9	1.0749	1.0362	1.2371	1.0745
16	1.1810	1.0369	1.3446	1.3241
25	1.2229	1.0452	1.3281	1.1117
100	1.0429	1.0190	1.0551	1.1662

due to the increased collisions in the network as more nodes compete to transmit at the same time.

Another reason is that NS-2 implements a cumulative noise and back-off strategy. This means that when a Data Packet arrives at the destination node, its RSSI is compared with the cumulative noise level sensed by the node's radio and with a receive threshold. The receive threshold value influences the 802.11 MAC in such a way that any signal above the receive threshold is considered to be noise by MAC, causing MAC to defer its transmission and initiate back-off. As more secondary transmissions in the larger network sizes take place, the noise increases, substantially hindering packet reception. However; we observe that the proposed A-RTSCTS method always had better throughput performance than the S-RTSCTS method.

The throughput improvement comparisons from the full capacity scenarios are presented in Table 4. Similar to the 3 traffic flow scenarios, we observe the biggest gain in throughput by A-RTSCTS over S-RTSCTS in the simulation scenarios for 16 and 25 nodes. In case of mobility for 16 nodes, we observe a throughput gain of 34.46% and a 32.41% for the grid and random distribution respectively. We also observe good performance in the case of mobile nodes in a 25 node grid topology where throughput gain was 32.81%. Overall, the simulation scenarios involving the topology sizes with a 100 nodes had the worst throughput, with gains as low as 1.9% in the case of static random nodes. The lack of throughput gain between A-RTSCTS and S-RTSCTS in the case of 100 nodes can be attributed to increased collisions from many more parallel transmissions.

$$Improv_{Ratio} = \frac{\textit{A-RTSCTS throughput}}{\textit{S-RTSCTS throughput}} \tag{3}$$

Since the performance of our method largely depends on the routing path selected by the AODV routing protocol, the improvement it provides is largely depended on how far the next-hop node in the selected path is located and the number of exposed nodes or hidden nodes included within that transmission range. If the selected path requires a next-hop node that has significant distance between it and the source node, more exposed nodes will be denied the opportunity to transmit and the A-RTSCTS's performance is almost similar to that of S-RTSCTS.

In Figs. 16 and 17, we show the ratio of dropped RTS packets for both A-RTSCTS and S-RTSCTS in relation to sent RTS packets, in static and mobile grid topologies. This is to deduce the effect of exposed nodes in the network when using either of the two methods. For both static and mobile topologies, we observe that the S-RTSCTS method experiences a larger ratio of dropped RTS packets, almost double that of the A-RTSCTS method.

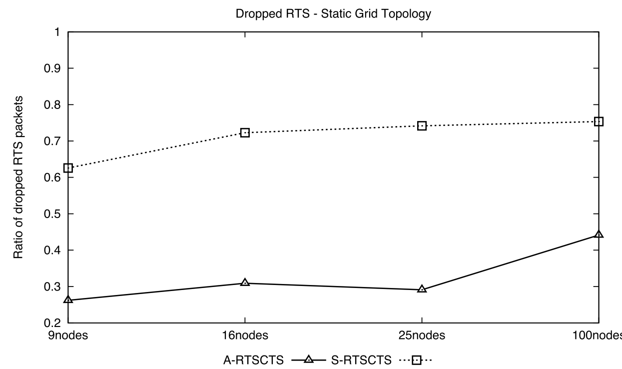


Fig. 16 Ratio of dropped RTS in static grid topologies.

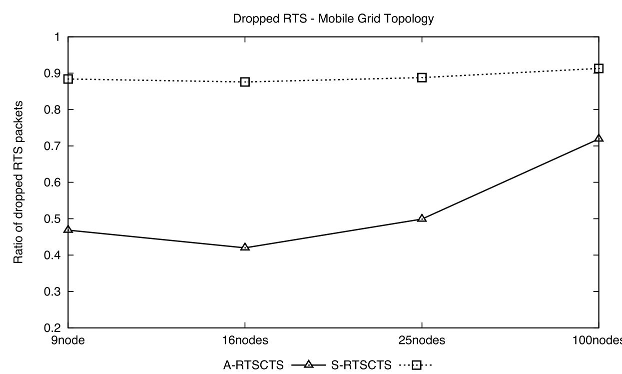


Fig. 17 Ratio of dropped RTS in mobile grid topologies.

In comparison to S-RTSCTS, the difference in dropped RTS packets is larger in cases of 16 and 25 nodes in Figs. 16 and 17. This leads to the higher throughput gains for A-RTSCTS over S-RTSCTS observed in 16 and 25 nodes. However, in the case of 9 nodes, we assume a significant number of the nodes in the network are included in the transmitting node's RTS range even when using A-RTSCTS. These nodes thus set NAV and reject incoming requests for communication which leads to the dropped RTS packets. Hence, the difference in dropped RTS packets between A-RTSCTS and S-RTSCTS is not so significant. Similarly, the 100 nodes topology experiences severe interference as the number of flows increase hence the difference in dropped RTS packets between the two methods is again less.

RTS packets dropped in the NS-2 simulation reported the reason as MAC_BUSY meaning the MAC retry count was exceeded (after 7 failed RTS transmissions). For S-RTSCTS, more nodes are included in the RTS transmission range than in A-RTSCTS. The nodes in the S-RTSCTS method thus enter the NAV period and back-off from accessing the channel, and thus drop any incoming requests for communication.

Table 5 Simulation parameters: PMAC vs A-RTSCTS.

Parameters	PMAC	A-RTSCTS
Number of nodes	14	14
RTS/CTS rate	5.5Mbps	1Mbps
DATA rate	24Mbps	24Mbps
ACK rate	24Mbps	1Mbps
Carrier sensing range	250m	250m
Simulation time	50s	50s

Table 6 Simulation results:PMAC vs A-RTSCTS

Topology	PMAC Average throughput (bytes/sec)	A-RTSCTS Average throughput (bytes/sec)
Multi-hop	2846.451	8617.86
9nodes Grid	43950.443	46781.677

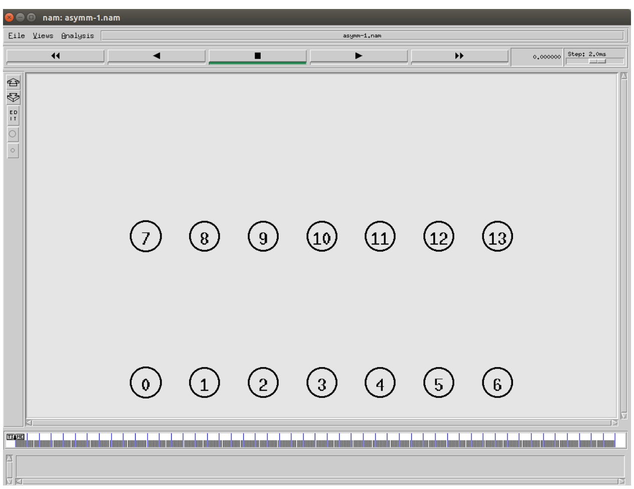


Fig. 18 Multi-hop chain topology.

4.3 Comparison with PMAC

We compare the performance of our proposed method to the PMAC method proposed in [15]. PMAC is a good candidate for comparison to our proposed method because, similar to our method, PMAC also uses information recorded from the initial transmission of RTS and CTS packets to influence the transmission of subsequent DATA packets. In PMAC, after the RTS/CTS frames are exchanged, the source node uses the received power of the CTS frame to estimate the transmit power required for the reliable reception of the subsequent DATA packet. ACK frames are sent using RTS' received power. This is done on a per packet basis. For the initial comparison, we use the same chain topology as that used in [15]. The simulation parameters are given in Table 5.

Table 6 shows the results from the comparisons of A-RTSCTS to PMAC using the multi-hop chain topology shown in Fig. 18, used by the authors in [15]. Additionally, we compare the performance of PMAC to A-RTSCTS in case of a 9 node grid topology used in our paper. From the results, we observe a significant difference in average throughput, especially in the multi-chain topology. Since the transmission rate change and the power control in PMAC are done on a per packet basis, once the the parameters for transmission are decided, they do not change until the transmission to the destination node finishes. With A-RTSCTS on the other hand, the link to each node has a different RTS transmission range which is dynamically changed as the transmission progresses. Hence the cross-layer design of our method has a significant effect on throughput.

Another concern is that PMAC only adjusts the transmit power for DATA and ACK packets but RTS/CTS packets are still sent using the normal transmit power. This means that exposed nodes that receive the RTS packet will still set NAV and thus still back-off from accessing the channel. The approach by PMAC blocks a large number of potentially beneficial transmissions.

5. Considerations

This paper goes a step further in evaluating the Asymmetric RTS/CTS idea by considering a cross-layer design to implement end-to-end, multi-hop communication. Rather than depend on a simplified adjustment of the RTS transmission range using pre-determined ranges, we consider an actual wireless network environment where transmission ranges between wireless nodes can largely not be pre-determined. We thus use the AODV routing protocol to determine the routing path that a Data packet has to take depending on the lowest hop count to the destination node.

In modelling the RTS transmission range to follow a next-hop node's RSSI, we had to make some simplifying assumptions. In our proposal, periodic HELLO messages are used for route maintenance and to discover new neighbours within a set HELLO_INTERVAL time. Continually broadcasting HELLO messages helps our strategy to work effectively, however an increased number of HELLO messages consumes the network resources and bandwidth. In our simulation, we assumed this possibility to be negligible, without a significant impact on overall network throughput. Consequently, turning off HELLO messages means some node's RSSI may not be recorded in time which may introduce further delays and lead to the failure of the proposed method.

Another assumption is that in a multi-hop mobile node communication scenario, there may be frequent link failures because of rapid changes in topology due to node mobility. The next-hop neighbour might move away in the time that the source node sets its RTS transmission range to its RSSI level. To accommodate this possibility, we include a margin in our simulations to account for such errors. This allows the nodes in our simulation scenarios to maintain connectivity even at mobility speed = 2 metres per second.

While the evaluation results mainly focus on throughput gain for A-RTSCTS over S-RTSCTS, the results obtained are satisfactory in assessing the proposed method's performance. In future works, we will consider evaluation criteria such as CBR packet loss, delay and AODV packet overhead.

6. Conclusion

Hidden and exposed nodes in wireless networks continue to characterize the IEEE 802.11 DCF MAC protocol. Using virtual carrier sensing, which invokes the RTS/CTS handshake, roles of the source and the destination nodes are interchanged several times during the communication process. This means that neighbouring nodes of both these nodes should be silent so as to not cause interferences with the ongoing communication. However; the RTS/CTS handshake that solves the hidden node problem also introduces the exposed node problem. Nevertheless, many researchers have suggested that exposed nodes are able to transmit to other nodes in their transmission ranges without harming an ongoing transmission [3], [7], [8].

In this paper, we evaluate improvements made to a method for reducing exposed nodes in IEEE 802.11 WLANs using asymmetric transmission ranges for RTS and CTS frames. Our method builds on the Asymmetric RTS/CTS method introduced in [5] to consider end-to-end, multi-hop communication by using a cross-layer design between Layer 2 and Layer 3. Using RSSI information recorded during the AODV path discovery process, we set RTS transmit power to be equivalent to the next-hop node's RSSI. This allows us to utilize the shortest route to the destination node while keeping the RTS transmission range at a minimum. Simulation results in NS-2 show that the improved Asymmetric RTS/CTS method (i.e A-RTSCTS) has better overall network throughput than the Standard RTS/CTS method (i.e S-RTSCTS) with throughput gains of up to 30%. We also show that A-RTSCTS performs better than an existing method.

Our evaluations used simplifying, general assumptions i.e inferring the RTS transmit power from a next-hop node's RSSI value. However, RSSI values may be susceptible to the influence of the physical environment which can lead to some estimation errors. Future work will consider other localization algorithms for obtaining the next-hop node's transmission range. In addition to average network throughput, we will also consider other evaluation criteria such as CBR packet loss, delay and AODV packet overhead.

Acknowledgments

The authors would like to thank Dr Akihisa Matoba for his assistance with this research.

References

[1] F. Gebali, Analysis of Computer Networks, Springer, 2015.
[2] X. Yang and N.H. Vaidya, "On the physical carrier sense in wireless ad hoc networks," Proc. IEEE Infocom, March 2005.
[3] S. Chakraborty, S. Nandi, and S. Chattopadhyay, "Alleviating hidden and exposed nodes in high-throughput wireless mesh networks," IEEE Trans. Wireless Commun., vol.15, no.2, pp.928–937, 2016.
[4] K. Xu, M. Gerla, and S. Bae, "Effectiveness of RTS/CTS handshake in IEEE 802.11 based ad hoc networks," Ad Hoc Networks Journal, vol.1, no.1, pp.107–123, Elsevier, July 2003
[5] A. Matoba, M. Hanada, H. Kanemitsu, and M.W. Kim, "Asymmetric RTS/CTS for exposed node reduction in IEEE 802.11 ad hoc networks," JCSE, vol.8, no.2, pp.107–118, 2013.
[6] H. Zimmermann, "OSI reference model-The ISO model of architecture for open systems interconnection," IEEE Trans. Commun., vol.28, no.4, pp.425–432, April, 1980.
[7] D. Shukla, L. Chandran-Wadia, and S. Iyer, "Mitigating the exposed node problem in IEEE 802.11 ad hoc networks," Proc. 12th ICCCN, pp.157–162, Oct. 2003.
[8] P. van de Ven, A.J. Janssen, and J. van Leeuwaarden, "Balancing exposed and hidden nodes in linear wireless networks," IEEE/ACM Trans. Netw. (TON), vol.22, no.5, pp.1429–1443, 2014.
[9] S. Rackley, Wireless Networking Technology: From Principles to Successful Implementation, Oxford, Newnes (Elsevier), 2007.
[10] Z.J. Haas and J. Deng, "Dual busy tone multiple access (DBTMA)-a multiple access control scheme for ad hoc networks," IEEE Trans. Commun., vol.50, no.6, pp.975–985, 2002.
[11] J. So and N.H. Vaidya, "Multi-channel mac for ad hoc networks," Proc. 5th ACM MobiHoc, 2004.
[12] L. Jiang and S.C. Liew, "Improving throughput and fairness by reducing exposed and hidden nodes in 802.11 networks," IEEE Trans. Mobile Comput., vol.7, no.1, pp.34–49, 2008.
[13] P. Karn, "MACA-A new channel access method for packet radio," Proc. ARRL/CRRL pp.134–140, 1990.
[14] R. Hunjet, "A capacity enhancing modification to RTS/CTS," 2015 22nd International Conference on Telecommunications (ICT), pp.67–73, Sydney, NSW, 2015.
[15] C. Huang, C. Lea, and A.K. Wong, "The hidden and exposed terminal problem," Computer Networks, vol.56, no.14, pp.3261–3273, 2012
[16] A. Butala and L. Tong, "Cross-layer design for medium access control in CDMA ad-hoc networks," EURASIP J. Applied Signal Processing, pp.129–143, Feb. 2005.
[17] T. Tsuboi, T. Ito, C. Hitsu, and H. Ueda, "Proposed multi-channel ad hoc network with cross-layer design of MAC and routing protocols," Proc. APSITT, pp.1–3, Colombo, 2015.
[18] Cisco, "Cisco Aironet 802.11a/b/g Wireless LAN Client Adapters (CB21AG and PI21AG) Installation and Configuration Guide, OL-4211-01," Cisco, https://www.cisco.com/c/en/us/td/docs/wireless/wlan.html, accessed Feb. 2. 2017.
[19] M. Cui and V.R. Syrotiuk, "Cooperative signalling and its application in a power-controlled MAC protocol," Proc. AdHoc Now'09, Murcia, Spain, Sept. 2009.
[20] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (AODV) routing," IETF RFC 3561, July 2003.
[21] A.S. Kim, J. Hwang, and J. Park, "Enhanced indoor positioning algorithm using WLAN RSSI measurements considering the relative position information of AP configuration," Journal of Institute of Control, Robotics and Systems, vol.19, no.2, pp.146–151, 2013.
[22] T. Issariyakul and E. Hossain, Introduction to Network Simulator NS2, Springer US, Boston, MA, 2012.
[23] C.-K. Toh, Ad Hoc Mobile Wireless Networks: Protocols and Systems, Prentice Hall PTR, Englewood Cliffs, NJ, 2002.
[24] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," IEEE J. Sel. Areas Commun., vol.18, no.3, pp.535–547, 2000.
[25] M. Sarker and M. Uddin, "Saturation throughput analysis of a carrier sensing based MU-MIMO MAC protocol in a WLAN under fading and shadowing," Wirel. Netw., pp.1–18, Springer Nature, 2017.
[26] J.F. Kurose and K.W. Ross, Computer Networking: A top-down approach, Pearson, Boston, 2013.
[27] C. Bettstetter, H. Hartenstein, and X. Perez-Costa, "Stochastic properties of the random waypoint mobility model," ACM/Kluwer Wirel. Netw., vol.10, no.5, pp.555–567, Sept. 2004.

Emilia Weyulu received her B.Sc. degree in Computer Science from the University of Namibia in 2011. From 2011 to 2015, she was a network technician with Telecom Namibia. She is currently a Masters student with the Graduate school of Informatics at the Tokyo University of Information Sciences in Chiba, Japan. Her research interests include Ad Hoc Wireless LANs and QoS. Ms Weyulu is currently a student member of IEEE.

Masaki Hanada received a B.E. degree in Resources Engineering from Waseda University in 1996, and M.S. and D.S. degrees in Global Information and Telecommunication Studies from Waseda University in 2003 and 2007, respectively. He is an associate professor in the Department of Information Systems, Tokyo University of Information Sciences. His research interests include QoS/traffic control and resource management in communication networks.

Hidehiro Kanemitsu received the B.S. degree in science from Waseda University, Japan, and the M.S. and D.S. degrees in global information and telecommunication studies from Waseda University, Japan. His research interests include parallel and distributed computing, grid, peer-to-peer computing, and web service technology. He is currently a Senior Assistant Professor at the School of Computer Science, Tokyo University of Technology, Japan. He is a member of IEICE, IPSJ, and IEEE.

Eun-Chan Park received B.S., M.S., and Ph.D. degrees from the School of Electrical Engineering and Computer Science, Seoul National University, Seoul, Korea, in 1999, 2001, 2006, respectively. From 2006 to 2008, he was a senior engineer with Samsung Electronics, Korea. He is currently an associate professor in the Department of Information and Communication Engineering, Dongguk University, Seoul since 2006. His research interests include network performance analysis, resource allocation and MAC protocol design of wireless networks.

Moo Wan Kim received B.E., M.E. and Ph.D. degree in electronic engineering from Osaka University, Osaka, Japan in 1974, 1977 and 1980, respectively. He joined Fujitsu Lab. in 1980 and had been engaged in research and development on multimedia communication systems, Intelligent Network, ATM switching system and operating system. In 1998 he joined Motorola Japan and had been engaged in research and development on CDMA2000 system. In 2000 he joined Lucent Japan and had been engaged in research and development on W-CDMA system, IMS and Parlay. In 2005 he joined Tokyo University of Information Sciences and has been engaged in research on Ubiquitous Network.