ISLAMIC UNIVERSITY OF TECHNOLOGY

RARRA
Receiver Assisted Robust Rate Adaptation

Authors:

Nafiul Rashid (094415)

Syed Sabir Salman-Al-Musawi (094406)

Supervisor:

Dr. Muhammad Mahbub Alam

Associate Professor

Department of Computer Science and Engineering

A thesis submitted to the Department of CSE
in partial fulfilment of the requirements for the degree of
B.Sc. Engineering in CSE

Academic Year: 2012-2013

A Subsidiary Organ of the Organization of Islamic Cooperation
Dhaka, Bangladesh

October 2013

Declaration of Authorship

This is to certify that the work presented in this thesis is the outcome of the analysis and investigation carried out by Nafiul Rashid and Syed Sabir Salman-Al-Musawi under the supervision of Dr. Muhammad Mahbub Alam in the Department of Computer Science and Engineering (CSE), IUT, Dhaka, Bangladesh. It is also declared that neither of this thesis nor any part of this thesis has been submitted anywhere else for any degree or diploma. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given.

Authors:

Nafiul Rashid
Student ID - 094415

Syed Sabir Salman-Al-Musawi
Student ID - 094406

Supervisor:

Dr. Muhammad Mahbub Alam
Associate Professor
Department of Computer Science and Engineering
Islamic University of Technology (IUT)

Abstract

The IEEE 802.11 wireless local area network (WLAN) standard, especially 802.11a remains the most popular way to exchange data over wireless links. The major requirement is to adapt to highly dynamic channel conditions with minimum overhead and ensure robustness and speed of transmission. To this end we propose a novel Rate Adaption Scheme RARRA (Receiver Assisted Robust Rate Adaptation). Our key contributions include exploiting the more precise channel estimation of SNR-based Rate Adaptation coupled with estimating the channel condition at the receiver and finally sending this estimated information to the transmitter with minimum overhead. In other words we avoid RTS/CTS overhead to send the channel condition to the transmitter and use acknowledgment rates to serve' this purpose. Secondly, we differentiate the cause of frame loss as either due to channel error or collision using RTS/CTS but in an adaptive fashion to minimize overhead but at the same time ensure that rate is not falsely changed due to frame loss caused by collision. RARRA exploits the best of SNR based approaches and provides channel condition at the receiver to the transmitter with minimum overhead thereby ensuring optimal rate switching decision aided by Adaptive RTS that provides robustness.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

The IEEE 802.11 wireless local area network (WLAN) standard remains the most popular way to exchange data over wireless links. One of the fundamental problems of any wireless technology is the volatile nature of the channel, which requires adaptation to its time-changing properties. To this end, the 802.11 standard defines a set of modulation and coding schemes (MCS), each of which has a nominal transmission rate that allows a trade-off between robustness and speed of the transmission; hence the term rate adaptation(RA). However, there is no single standardized system in place to adapt to the most efficient rate at any given point in time. Instead, numerous RA algorithms have been proposed over the years. One fundamental problem of rate adaptation is scarcity of information. The sender needs to adapt the transmission rate; however, the information about reception quality is only available at the receiver, and needs to be fed back to the sender by some means.

1.1.1 What is rate adaptation?

o The method used to dynamically select the transmission rate of wireless networks based on time-varying channel quality. Rate adaptation affects throughput performance and should be adjusted by channel condition. Also known as "RA".
o It is the roadmap of a successful adaptive solution and answers the following questions:-

- What to adapt to?

- How to adapt?
- How well it can adapt?
- What should an adaptive solution adapt to?

o A rate adaptation algorithm should identify each possible scenario and handle each one by one. It is the method to select the transmission rate in real time. Rate adaptation affects throughput performance and should be adjusted by channel condition.

1.1.2 What are the different standard rates?

Standards used in WLANs include the following:

Table: 1.1 gives us Standards used in WLANs

Table 1.1: Different WLAN standards and their supported rates

Standards	**Supported Rates/(Mbps)**
802.11a (8 rate options)	6 , 9 , 12 , 18 , 24 , 36 , 48 , 54
802.11b (4 rate options)	1, 2 , 5.5 , 11
802.11g (12 rate options)	11a set + 11b set

802.11 a/b/g standards allow for the use of multiple transmission rates. These rates as per standards are as follows. Among them 802.11a is the most widely used and deployed.

1.1.3 Example of Rate Adaptation

Here in the figure: 1.1 the transmission rate should be adjusted according to the channel condition. Here it is apparent that when the channel condition is good a rate adaptation algorithm should be able to increase rate to benefit the good channel quality and do the opposite when the channel quality worsens.

Table: 1.2 shows the effect of very high and low rates

Table 1.2: Effect of very high and low rates

Rate Too High	**Rate Too Low**
Increases Loss Ratio	Capacity Under-Utilized
Decreased Throughput	Decreased Throughput

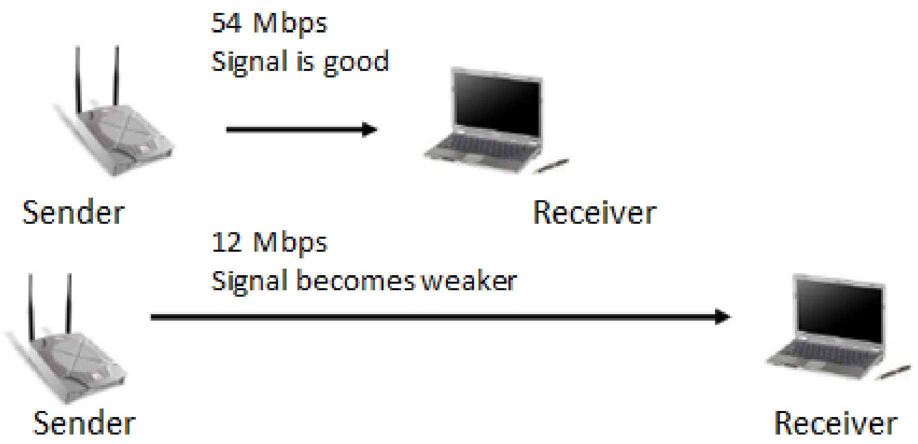


FIGURE 1.1: channel conditions in different scenerio

1.1.4 Importance of Rate Adaptation

Rate adaptation plays a critical role to the throughput performance.

- When the rate is high but channel quality is poor, loss ratio increases and through put decreases.
- When the rate is low but channel quality is very good, it leads to under-utilization of channel quality and hence low throughput.
- Rate adaptation affects the throughput performance!

Hence the job of Rate Adaptation is to better exploit the Physical Layer multi-rate capabilities and adjust data rate in line with channel quality to maximize throughput and channel utilization.

1.1.5 Classification of Rate Adaptation approaches

There are mainly two types of approaches:-

- SNR based/Best RAs.
- Frame based/Loss based RAs.

SNR based RAs:

- Uses physical layer metric i.e., SNR(Signal to Noise Ratio) values to estimate channel quality.

- SNR-based designs translate the measured SNR into a transmission rate based on predefined mappings.
- It has the problem of hardware compliance, therefore less widely used.

Frame based RAs:

- Uses link layer metric i.e., consecutive success/losses to estimate channel quality.
- Loss-based designs estimate the channel quality based on the outcome of previously transmitted frames.
- This approach is widely used and is hardware compliant.
- It has the problem of rate under selection due to its sequential nature of rate adaptation. So it cannot utilize channel variation and switch to the optimal rate.

So we will try to develop such an algorithm that can exploit the best of the two approaches, i.e., it will be hardware compliant and can switch to the optimal data rate.

1.1.6 Pros and Cons of Rate Adaptation approaches

SNR based RAs:-

Pros: The biggest advantage of SNR-based RA is that it can switch to the optimal rate which yields the best performance in terms of throughput. This is the case because it measures the physical layer metrics i.e., SNR to estimate the channel condition based on predefined mappings. It causes good channel utilization.
Cons: It has the problem of hardware compliance, SNR calibration and is therefore less widely used in wireless network drivers.

Frame based RAs:-

Pros: It does not have the problem of hardware compliance, SNR calibration and is therefore more widely used in wireless network drivers and deployed very much commercially.
Cons: It estimates channel condition based on previously transmitted frames. Due to the sequential approach of rate change i.e., switch one rate option at a time every time it is essential to change data rate, we get channel underutilization and rate under selection. The data rate fails to switch in line with the dynamic changes in channel conditions hence channel utilization is not optimum.

1.1.7 Different rate switching techniques

- **Sequential:**It switches to next higher/lower rate based on channel quality. E.g.-if the 802.11a standard supports rates 6, 9, 12, 18, 24, 36, 48 and 54.If the current rate is 18 and channel degrades, the rate falls to 12. It leads to underutilization of the channel capacity.
- **Optimal:**It switches to the optimal rate based on channel quality. E.g.-if the 802.11a standard supports rates 6, 9, 12, 18, 24, 36, 48 and 54.If the current rate is 18 and channel degrades and the current channel supports 6 Mbps, the rate falls to 6 directly. It leads to optimal utilization of the channel capacity.
- **Random:**It switches to the higher/lower rate randomly based on channel quality. E.g.-if the 802.11a standard supports rates 6, 9, 12, 18, 24, 36, 48 and 54.If the current rate is 18 and channel improves, the rate rises randomly to one of the higher rates. It leads to improper utilization of the channel capacity.

1.1.8 Rate Avalanche Effect

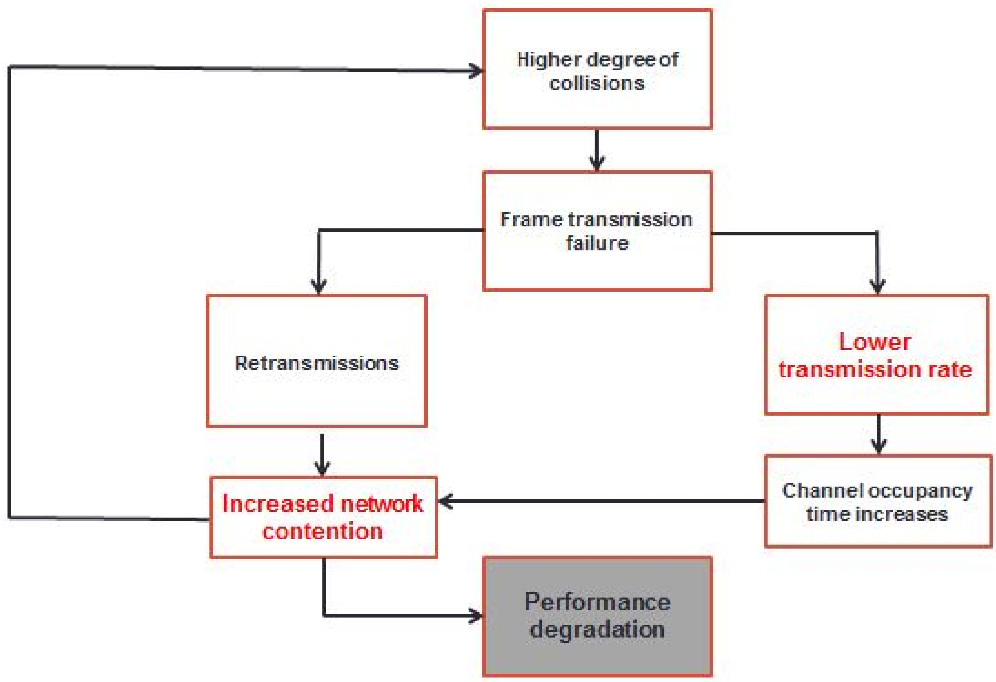


FIGURE 1.2: Effect of Rate Avalanche

Here in the figure: 1.2 it discusses a very important issue i.e., the rate avalanche effect. The phenomena is summarized as follows:-

- When there is high network congestion and packets are getting dropped, one of two options is available, either Retransmit or Lower data rate.
- Both retransmission and lowering the data rate increases channel occupancy time.
- Each node occupies the channel for a longer time either by transmitting the same frame again and again or by sending at a low rate.
- This further increases channel contention and leads to even higher network congestion.
- Hence a vicious cycle exists.
- It is the main cause we introduce the concept of RTS/CTS in rate adaptation to differentiate the cause of frame loss and not decrease rate when packets are lost due to congestion.

Chapter 2

Literature Review

2.1 Related Work

Physical rate adaptation in IEEE 802.11 is a well-known and deeply studied issue. Algorithms have been proposed in the literature and part of them cannot be implemented in the real network interfaces because they are not standard compliant. In this section, we describe the most known rate adaptation algorithms, bringing more details to the ones we compare with our algorithm.

2.1.1 ARF(Auto Rate Fallback)

ARF[1] is a widely adopted and well known rate adaptation algorithm. The decision whether to increase or decrease the transmission rate is based on the number of consecutive successfully or unsuccessfully transmission attempts, respectively. In other words ARF increases rate on 10 consecutive successes and decreases rate on 2 consecutive failures.
This algorithm is widely adopted because it is simple. The main problem of this algorithm is that it cannot distinguish between losses due to collision from losses due to channel, so it achieves poor performance in multi-user scenarios. Another problem, pointed out in is that it tries a higher rate every time it obtains fixed number of successfully transmission attempts, even if the current rate is the most convenient. To alleviate this problem, the authors of proposed the Adaptive ARF (AARF) algorithm that behaves like ARF with the difference that the number of consecutive successfully transmission attempts before trying the higher rate is incremented exponentially every time the higher rate transmission fails. AARF performs better than ARF in case of single-user scenarios, but it has the same problems as ARF in multi-user scenarios.

2.1.2 SampleRate

SampleRate[2] sends packets at the bit-rate that has the smallest average packet transmission time as measured by recent samples. A key aspect of the design of SampleRate is the way it periodically sends packets at bit-rates other than the current bit-rate to estimate their average transmission time.
The algorithm works as follows:

- If no packets have been successfully acknowledged, return the highest bit-rate that has not had 4 successive failures.
- Increment the number of packets sent over the link.
- If the number of packets sent over the link is a multiple of ten, select a random bit-rate from the bit-rates that have not failed four successive times and that have a minimum packet transmission time lower than the current bit-rate's average transmission time.
- Otherwise, send the packet at the bit-rate that has the lowest average transmission time.

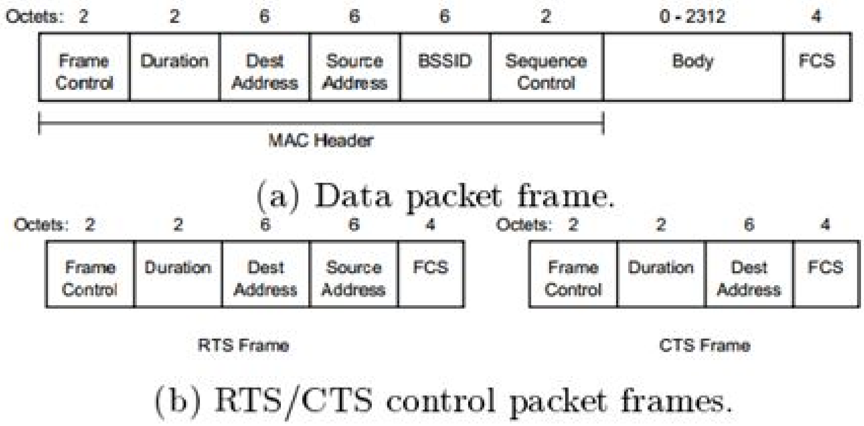
In brief, SampleRate starts transmission at highest rate. It Decrease to next lower rate on 4 consecutive failures and on 10 consecutive successes it randomly choose from the higher rates.

2.1.3 RBAR(Receiver Based AutoRate)

RBAR[3] is one of the earliest SNR based protocols. The novelty of RBAR is that its rate adaptation mechanism is in the receiver instead of in the sender. Hence it exploits the receiver side channel conditions to make rate decisions. Its key features are:

- Sender sends RTS message before every transmission to the receiver.
- In RTS frame instead of carrying 16 bit "duration field", it carries 4 bit "rate field" and 12 bit "length field" (here length means packet size).
- The same is for CTS frame.
- Thus neighbors can calculate the duration for NAV from this two fields "rate & length".

- Receiver measures the RSSI of the RTS frame received.
- Depending on that RSSI receiver sends the CTS frame to the sender telling about the next data rate expected in the rate field of CTS frame.
- The length field of CTS frame contains the packet size of the CTS frame.

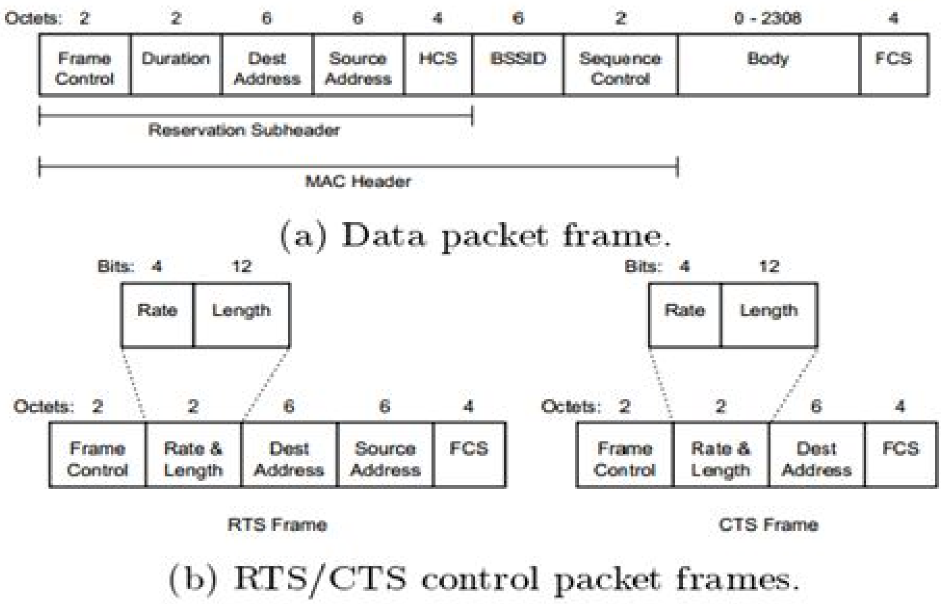


(a) Data packet frame.

(b) RTS/CTS control packet frames.

FIGURE 2.1: Standard MAC Frame format in IEEE 802.11

The above figure represents the conventional MAC frame formats used in IEEE 802.11 for wireless networks. Below is the figure of the MAC and physical layer formats used in the RBAR protocol.



(a) Data packet frame.

(b) RTS/CTS control packet frames.

FIGURE 2.2: MAC and Physical Layer Frame format in RBAR

2.1.4 CHARM(Channel-aware Rate Adaptation Algorithm)

Unlike traditional probe-based or frame based rate adaptations like ARF and AARF which uses consecutive success/loss information CHARM[4] is purely SNR-based that uses Channel reciprocity to estimate path loss and use accurate channel measurements for rate selection without using RTS/CTS.CHARM is suitable for dynamic and uses RSSI measurements at the receiver to estimate RSS, the most important component of SINR. The contribution of CHARM is four-fold:

1) **PATH LOSS MONITORING:** RSSI is a good approximation of RSS and, combined with transmit power information , it can be used to estimate path loss at receiver. CHARM obtains path loss information by leveraging the Reciprocity Theorem which states that, "the instantaneous path loss between two nodes is the same in both directions and a transmitter can obtain the path loss to a receiver by measuring the path loss from the receiver to the transmitter". CHARM continuously passively monitors the packets sent by any destinations of interest. The transmitter records the RSSI of the packets it overhears, and uses the RSSI to estimate the instantaneous path loss of the channel from the destination to itself .Due to channel reciprocity, this is also the instantaneous path loss to that destination. All path loss estimates are stored in a table with a timestamp for use by the path loss prediction algorithm.

2) **PATH LOSS PREDICTION OF FUTURE RECEIVED PACKETS:** A time-aware prediction algorithm assigns a weight to each received RSSI as they are measured and after every arrival from a specific host, the average RSSI of future received packets from that host is estimated. However it is important to consider the timing of the samples. Specifically, recent samples are more likely to be representative of the current channel conditions than older samples, so they should carry more weight.

$$RSSIAvg = \frac{(RSSIAvg * f(dT)) + RSSICur}{1 + f(dT)}$$

Here dT is the arrival time between two packets. f (dT) is a linearly decreasing function of dT , starting at 1 and decreasing to 0 when dT exceeds a decision time window . We use a window of two seconds since we did not observe any benefit from larger windows. In order to filter out large drops in RSSI that may not be representative of channel conditions CHARM considers a sharp drop of RSSI lasting for a single packet as due to noise and does not include it in average RSSI calculation. However, these drops if existing for more than a packet are counted.

3) **RATE SINR THRESHOLD ESTIMATION:** Once we know the average RSSI at every point of time updated by continuously received packet RSSI and past history of RSSI assigned weights according to time of arrival, we can define the minimum SINR threshold that allows a rate to be used such that packet can be decoded successfully at the receiver. The success at the predicted SINR obtained by long term average RSSI from above versus success at observed SINR is used to calibrate the threshold periodically. As these thresholds may vary from receiver to receiver, each transmitter contains a rate SINR threshold set for each receiver that it is communicating with, and updates these thresholds independently.

4) **RATE SELECTION:** Once we have the rate-SINR thresholds by obtaining path loss via channel reciprocity and hence estimating SINR we choose a set of transmission rates through lookup in a table with SINR thresholds for the intended receiver. Per packet the driver can specify several transmission rates, which will be used for the original transmission and each of the possible retransmissions. The multi-rate retry settings are used.

2.1.5 REACT(Rate Adaptation Using Coherence Time)

Key features:-

- This is SNR based approach.
- The receiver in REACT[5] informs the transmitter of the improved channel condition via altering the ACK transmission rate.
- The channel status information obtained via the preceding ACK frame will be valid for the following data frames.
- Because the channel coherence time in WLANs typically exceeds multiple frame transmission times.
- Upon receiving an ACK frame indicating the good channel condition, the transmitter increases the data rate to the next higher rate
- REACT identifies the reason of frame losses by exploiting the feed-back from the preceding ACK frame and the coherence time.
- After receiving an ACK frame indicating the improved channel condition, the transmitter can assume that the channel at the receiver will be favorable for the higher bit rate during the interval of the coherence time.

- Thus, the data frames that are lost during this interval are deemed to be lost due to occurrence of collisions, and not by channel errors.

When to increase the data rate:-

The 802.11 standard requires that the ACK frames be transmitted at the maximum bit rate that is constrained by two rules:
1) The transmission rate of an ACK frame should be less than or equal to that of the preceding data frame, and
2) The ACK frame is transmitted at a rate selected from the basic rate set. Here, 6 Mbps, 12 Mbps, 24 Mbps is the set of 802.11a mandatory data rates, so it was assumed to be the BSS basic rate set in this paper.

- ACK rate that conform to the above two rules the **legacy ACK rate**.
- The receiver, however, can transmit an ACK frame at a rate other than the legacy ACK rate, which is henceforth referred to as the **altered ACK rate**.
- There are **two possible options** for the altered ACK rate:-

1) **the next lower rate than the legacy ACK rate** :- is used when the data rate is faster than or equal to 12 Mbps

2) **the next higher rate than the legacy ACK rate** :- is used when the data rate is 6 or 9 Mbps.

When to decrease the data rate:-

- The key issue in rate decreasing is how to figure out the reason of frame losses that are due to channel errors or collisions.
- In order to cautiously differentiate frame losses, we exploit the coherence time in the wireless channel as follows:- A time-domain signal may be correlated over a certain amount of time, so that the channel does not experience a significant variation for the duration of the **coherence time** after receiving a channel status feedback.
- Once we calculate the coherence time we can figure out the causes for frame losses.
- The transmitter can figure out the reason of frame losses during that period as frame collisions instead of the reason being the bad channel condition.

- We call this time duration a **"green channel period"** during which stations do not suffer from frame losses due to the bad channel condition.
- The **green channel period** can help to adaptively use RTS probing.
- As because we have to use adaptive RTS probing ,if any frame losses occur after this green channel period.
- If the frame loss is due to channel condition, then two consecutive frame loss causes rate decrease.

2.2 Critiques On Related Works

Till now in our work we have encountered the above mentioned RA algorithms. However each one has its own good and bad sides. Several critiques have been put forward for each RA algorithm we have mentioned above. These are summarized below.

2.2.1 Frame Based or SNR Based

We are familiar that Frame Based approaches estimate channel condition based on previously transmitted frames. This use of link layer metrics causes rate under selection and channel underutilization.
ARF and SampleRate are two such frame based approaches each of which uses success/failure of previously transmitted frames and switches rate sequentially and randomly respectively. Hence they pose the disadvantages of traditional frame-based approaches.
On the other hand SNR based approaches which switch to optimal rate as governed by SNR as a measure of channel condition has optimal channel utilization. RBAR, CHARM and REACT obtain such benefits as being SNR-based approaches. They switch to optimal rate and use SNR as a physical layer metric for judging channel conditions.

2.2.2 Channel Quality Estimation at

RA algorithms usually make use of either link layer metrics i.e., previously transmitted frames or physical layer metrics such as SNR for estimating channel conditions. No matter what the metric is, the end where channel condition is measured is important. It is helpful to measure channel condition at the receiver since that

is the end where frames need to be received and decoded.
Channel condition measurement at the sender does not give us an accurate picture of channel conditions at the receiver since we cannot assume channel symmetry. Hence it is best estimated at the receiver where frames will be received and need to be decoded.
ARF and SampleRate measures channel condition at the sender. CHARM is also in the group but it uses Channel Reciprocity to assume a symmetric channel between sender and receiver which is practically infeasible. On the other hand RBAR and REACT estimates channel condition at the receiver which is good thing so it sends the best rate at which the data can be sent.

2.2.3 Rate Switching Techniques used

Sequential rate poses the problem of rate under selection while optimal rate switching improves to optimal rate selection. Random rate switching results in improper channel utilization.
ARF and REACT relies on sequential rate switching and so they only switch to the immediate higher or lower rate when channel conditions changes but this does not utilize dynamic channels which may suddenly improve or get worse.
However RBAR and CHARM uses optimal rate switching techniques. CHARM maps SNR values to data rates while RBAR uses the rate advertised by the CTS frame. Hence the rate is calibrated with the channel conditions.
Lastly, SampleRate increases rate randomly from a set of data rates higher than the current. But this random choice of rate leads to improper channel utilization.

2.2.4 Use of RTS/CTS

RTS (Request to Send) and CTS (Clear to Send) are control frames used for establishing connection between sender and receiver. The use of RTS occupies the channel and prevents collision from hidden terminals. However even though RTS/CTS ensures channel occupancy, it incurs overhead so its use should be minimized.
RBAR uses RTS/CTS always. It minimizes collision based losses because every transmission is guarded by RTS/CTS but incurs huge overhead and is unnecessary.
ARF, SampleRate and CHARM never use RTS/CTS which reduces overhead but increases vulnerability of collision based losses.
REACT on the other hand uses RTS/CTS in a different and most desirable fashion.

Not using RTS/CTS at all increases collision based losses and leads to inaccurate rate selection. Overusing RTS/CTS compensates the gain.
Hence an **Adaptive** approach is followed by REACT that uses RTS/CTS on demand. REACT incurs the marginal overhead with respect to RTS/CTS overhead for delivering the channel status information but changes the RTS window (number of RTS protected frames) based on **green channel period**.

2.2.5 Differentiating the Cause of Frame Loss

The rate avalanche effect is one of the main reasons why rate under selection degrades channel performance. Usually frame based RA algorithms experience it because they under select rates. It is important to differentiate between frame losses as either due to **collision** or **channel-error** because collision based losses falsely lower rates and degrades performance. The main use of RTS/CTS frame is to differentiate between the causes of frame loss.
ARF, SampleRate and CHARM never use RTS/CTS frames. Hence they are vulnerable to collision from hidden stations. Moreover they fail to differentiate the cause of packet loss and may falsely reduce rate due to collision based losses. This means such RA algorithms are likely to undergo the vicious cycle of the **rate avalanche effect**.
RBAR on the contrary uses RTS/CTS before every transmission and hence reduces collision based losses and prevents rate under selection. Naturally it can differentiate the cause of frame loss. RTS/CTS confirms channel occupation and a subsequent frame loss means it is due to channel error so rate can be decreased.
Lastly, REACT uses Adaptive RTS. It uses RTS/CTS on demand to differentiate the cause of frame loss and avoid inaccurate rate selection due to collision based losses. It exploits the benefit of RTS but uses it adaptively depending on the channel coherence time. An RTS window gives protection to only a few frames. So overhead is reduced but differentiation of cause is achieved.

Chapter 3

Proposed Method

3.1 Motivation

So far we have seen the existing algorithms, each of them meets specific criteria but does not fulfill all the criteria that determines an algorithm to be robust and optimal .We are highly motivated to focus in that point. That is our main motivation is to develop such an algorithm that fulfills all the criteria of a robust and optimal algorithm.
Keeping all these in mind we propose a robust and optimal algorithm with following objectives:

- **A SNR Based approach :** As SNR Based approach provides more precise estimation of channel quality due to use physical layer metric that is the SNR value.
- **Uses 802.11a Standard rates :** As 802.11a rates are widely used.
- **Channel condition is measured at receiver :** As channel condition is best measured at receiver.
- **Receiver informs transmitter without RTS/CTS overhead :** Receiver uses acknowledgement rate to inform the transmitter.
- **Differentiate the cause of frame loss :** Uses Adaptive RTS to differentiate the cause of frame loss.
- **Switch to Optimal rate :** Switches to optimal rate according to channel condition.

3.2 Methodology

To achieve the objectives described above we propose some methods. These methods are implemented both at receiver and sender side.Following sections will describe about these methods.

3.2.1 Receiver Side Mechanism

As already mentioned we use acknowledgement rate to inform the transmitter about the channel condition. To select the ACKrate (Acknowledge Rate) that will determine the following transmission rate by the sender we have to follow the following steps :

- We maintain a Table of SNR ranges that maps to the ACKrate called SNR-ACKrate lookup table.
- Then we determine the SNR value of the received data frame.
- We use estimated SNR value to find an ACKrate from the SNR-ACKrate lookup table.
- Then we send the ACK(Acknowledgement) at the selected ACKrate.

The SNR-ACKrate lookup table[6] is given below : This SNR-ACKrate lookup

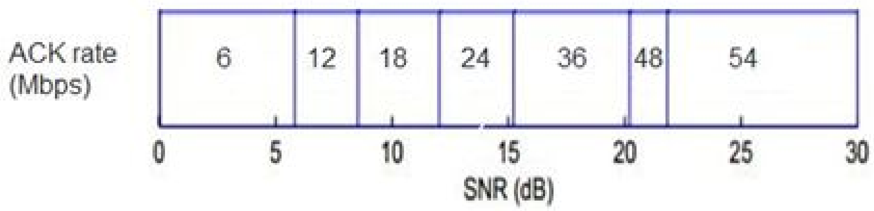


FIGURE 3.1: SNR-ACKrate Lookup table

table in [6] was implemented in the sender side but we will implement this table in the receiver side as channel condition is best measured at receiver side.
The Receiver side mechanism is demonstrated in the following Flowchart:

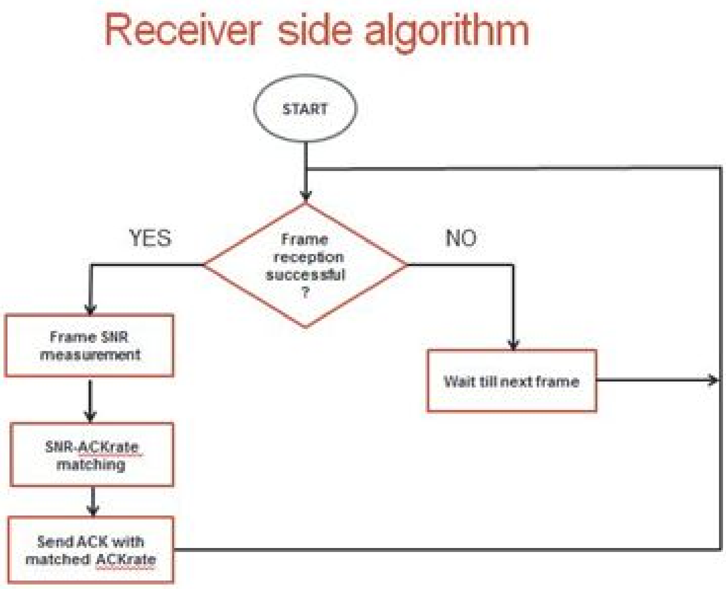


FIGURE 3.2: Receiver side algorithm of RARRA

3.2.2 Sender Side Mechanism

Though rate selection decision is made at the receiver side but data sending decision is made at the sender side.
The steps that are followed at the sender are:

- If ACK is received, send the next data frame at received ACKrate.
- If ACK is not received , retransmit the frame with RTS protection.
- Even after RTS protection , a frame loss indicates channel error because if it was due to collision then RTS use would result in successful transmission.
- In case of two consecutive failures, send the next frame at lowest rate.
- Because a successful transmission is very likely at lowest rate.
- Thus, once a successful transmission will return an ACK that tell appropriate channel condition channel condition through ACKrate.

The sender side mechanism is demonstrated in the following Flowchart:

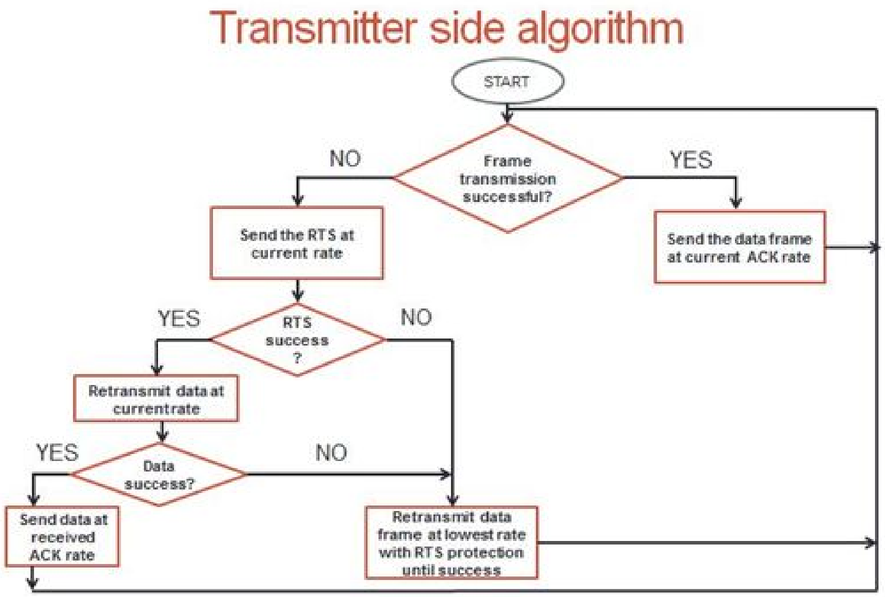


FIGURE 3.3: Sender side algorithm of RARRA

3.2.3 Use Of Adaptive RTS

The meaning of the term Adaptive RTS means using the RTS when it is necessary. Now the question comes when it is necessary?The simple answer is to find out the cause for frame losses.
The next question may arise why we need to find out the cause of frame losses?. Answer is frame loss may be due to collision or channel error.
Now when the frame loss is due to channel error then we can lower the rate immediately to improve the performance.But if the frame loss is due to collision then reduction of transmission rate will worsen the condition .That we call the Rate Avalanche Effect [Sec 1.1.8] that is already described .So to avoid this we have to use the RTS.
And our proposed method will use this Adaptive RTS to avoid the Rate Avalanche Effect. Thus it becomes more robust.

3.2.4 Operation Example

Now we go into the detail of the rationale of our proposed method. We will compare the rationale of our proposed method with one existing Frame based method ARF[1] and one existing SNR based method REACT[5].

In figure: 3.4 the ARF scheme underutilize the channel capacity due to sequential rate switching techniques. Suppose the channel now supports 18 Mbps while the sender sends at 48 Mbps. So to decrease to 18 Mbps it will get two consecutive failures at 48, 36, 24 then it will get to 18 then get successful transmission. Thus it will underutilize the channel condition.Similarly when the channel condition becomes suitable for 36 Mbps from 18 Mbps it will sequentially increase data rate every after 10 consecutive successful transmissions.Which will lead to the under-utilization of channel capacity.

Similarly in case of REACT as shown in figure: 3.5 scheme it will underutilize the channel capacity due to sequential rate switching techniques. It has the same limitations of rate decrease as in ARF. In case of rate increase it performs better than ARF but fails to switch to optimal rate directly. It performs poorly when channel condition fluctuates frequently.

Whereas in figure: 3.6 our proposed scheme RARRA provides optimal performance incase of rate decrease and rate increase. In case of rate decrease after two consecutive failures it sends the data at lowest rate which increases chance for packet to get through and get the acknowledgement. Once we get the acknowledgement we can know about the channel condition and switch to the optimal rate.

The diagrams therefore show the comparison of operation rationale .Here we compare the rationale of our proposed method with one existing Frame based method ARF[1] and one existing SNR based method REACT[5]

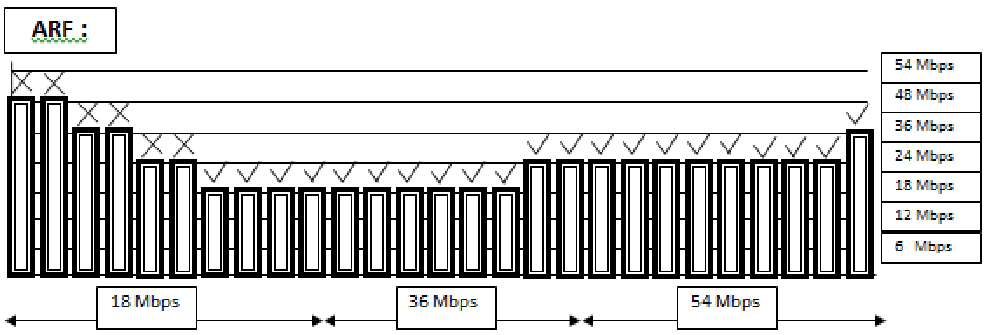


FIGURE 3.4: Operation rationale of ARF

3.2.5 Comparison Summary

Table: 3.1 gives the comparison of our proposed method with the existing rate adaptation algorithms

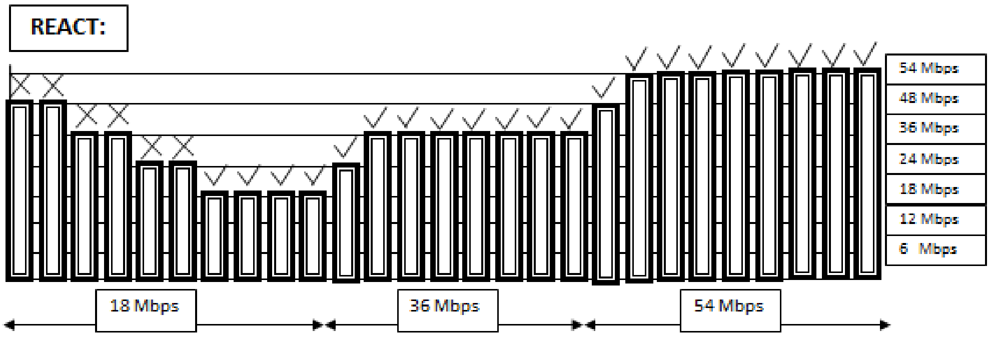


FIGURE 3.5: Operation rationale of REACT

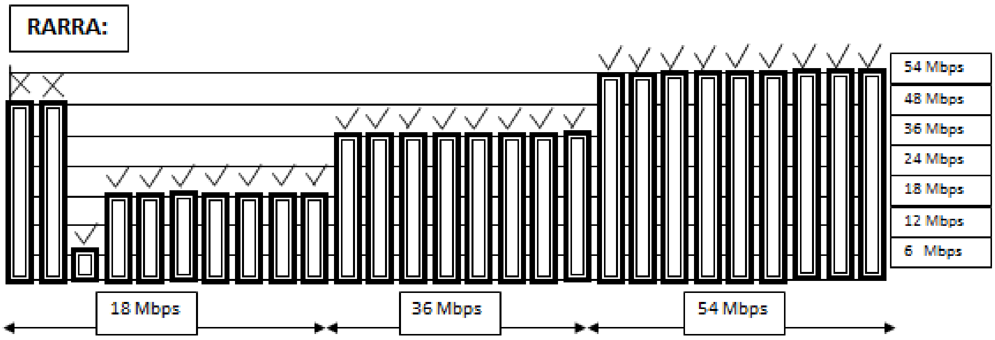


FIGURE 3.6: Operation rationale of RARRA

TABLE 3.1: Rate Adaptation Algorithms in a nut shell

Algorithm	Type	Switching Technique	Channel Quality at	RTS/CTS use
ARF	Frame Based	Sequential	Sender	Never
SampleRate	Frame Based	Random	Sender	Never
RBAR	SNR Based	Optimal	Receiver	Always
CHARM	SNR Based	Optimal	Sender	Never
REACT	SNR Based	Sequential	Receiver	Adaptive
RARRA	SNR Based	Optimal	Receiver	Adaptive

Chapter 4

Simulation Result and Performance Analysis

4.1 Simulation Setup Plan

- We will use ns-3 as our simulator.
- We will simulate in the indoor environments.
- For Large scale model: We will use log-distance path-loss model[7].
- For Small scale model: We will use Two ray ground propagation loss model.
- We will perform simulation on the following scheme:
  -ARF[1]
  -REACT[5]
  -RARRA

4.2 Results with various distances:

We will perform simulation on the proposed scheme RARRA along with ARF and REACT by varying distances on the following topology:

- **Type of network :** Adhoc
- **Number of nodes :** 2
- **Number of flows :** 1

- **Packet transmitted :** 100000
- **Packet size :** 2048 bytes
- **Flow data rate :** 54 Mbps
- **Path loss Model :** a) Log distance path loss model
  b) Two ray ground propagation loss model

- **Mobility Model :** a) Constant position mobility model
  b) Randomwalk2d mobility model

- **Initial Distance between nodes :** 5,10,15,20,25,30,35,40,45,50 meters

Now we evaluate the results of RARRA, REACT, ARF in various combination of the path loss and mobility model.

4.2.1 With various distances in "Log distance path loss model" and "Constant position mobility model":

RESULT TABLE: The result table is represented in figure: 4.1

	A(X)	B(Y)	C(Y)	D(Y)
Long Name	Distance	Throughput		
Units	m	Mbps		
Comments		RARRA	REACT	ARF
1	5	33.1094	32.7428	23.2648
2	10	30.5485	30.2315	23.2625
3	15	24.9426	24.7351	23.2603
4	20	24.9413	24.7337	23.258
5	25	24.9399	24.7324	22.7377
6	30	18.5093	18.0575	16.5452
7	35	18.5086	18.0568	17.3156
8	40	18.5079	18.0561	17.0744
9	45	18.5072	18.0555	13.5925
10	50	18.0721	17.6424	13.437

FIGURE 4.1: Table showing the results for different algorithms in "Log distance path loss model" and "Constant position mobility model"

COMPARISON GRAPH: The comparison graph is represented in figure 4.2

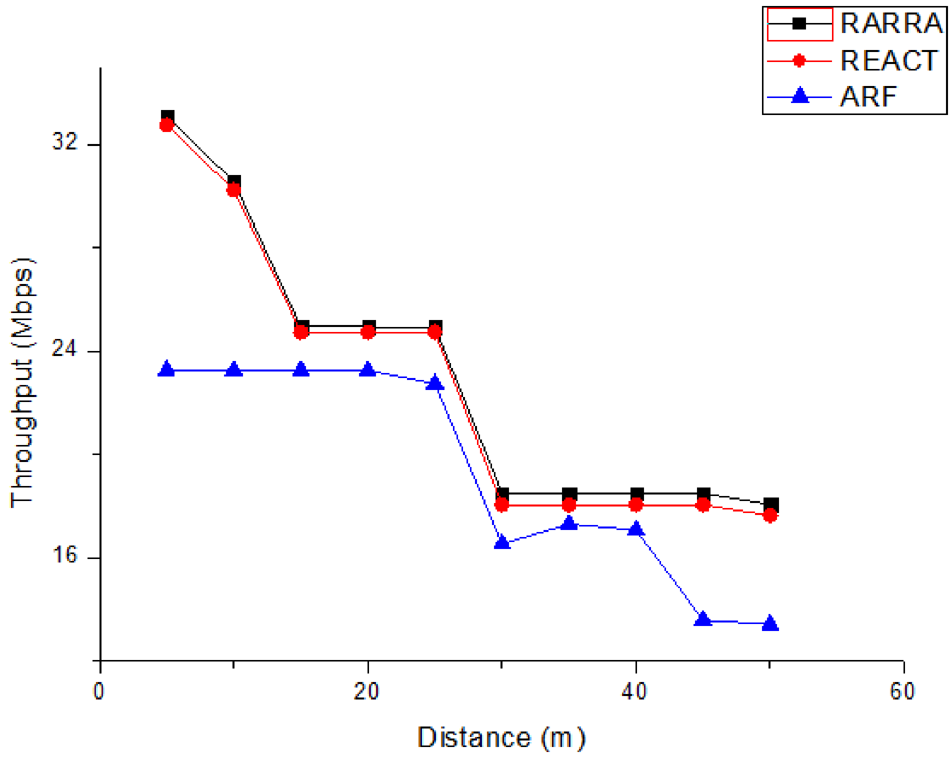


FIGURE 4.2: Graph showing the outcome for different algorithms in "Log distance path loss model" and "Constant position mobility model"

4.2.2 With various distances in "Log distance path loss model" and "Randomwalk2d mobility model":

RESULT TABLE: The result table is represented in figure: 4.3

COMPARISON GRAPH: The comparison graph is represented in figure 4.4

4.2.3 With various distances in "Two ray ground propagation loss model" and "Constant position mobility model":

RESULT TABLE: The result table is represented in figure: 4.5

COMPARISON GRAPH: The comparison graph is represented in figure 4.6

4.2.4 With various distances in "Two ray ground propagation loss model" and "Randomwalk2d mobility model":

RESULT TABLE: The result table is represented in figure: 4.7

COMPARISON GRAPH: The comparison graph is represented in figure 4.8

	A(X)	B(Y)	C(Y)	D(Y)
Long Name	Distance	Throughput		
Units	m	Mbps		
Comments		RARRA	REACT	ARF
1	5	33.1474	32.783	23.6714
2	10	30.5548	30.2458	23.669
3	15	24.965	24.7647	23.6604
4	20	24.9637	24.7634	23.6571
5	25	24.9624	24.7622	22.5683
6	30	18.5025	18.0729	17.5341
7	35	18.5008	18.0722	17.5166
8	40	18.5001	18.0715	17.3314
9	45	18.3094	18.0708	13.5866
10	50	17.9974	17.8945	13.6574

FIGURE 4.3: Table showing the results for different algorithms in "Log distance path loss model" and "Randomwalk2d mobility model"

From the above data that we collected, it is apparent that both REACT[5] and RARRA follow the same trend in terms of throughput changes. This is mainly due to the fact that we have used the same SNR range mapping in both algorithms to keep the conditions constant and vary only those parameters based on which comparisons are made. Here we have varied distance between nodes.

However, it is worth mentioning that our algorithm demonstrates a performance improvement in both static and mobile scenarios. We have used LogDistancePathLoss model and TwoRayGroundPropagationLoss model for both static and mobile scenarios.

For the static case, the changes are less apparent as channel conditions are more or less same. But in the dynamic case when nodes are mobile, channel condition varies much and it is in this case that our algorithm shows robustness.

The above figures for each of the path loss models clearly demonstrate that RARRA performs much better than REACT in mobile scenarios than in the static scenarios and ARF[1] is always at the bottom of the list whether nodes are static or dynamic.

4.3 Results with various Contending Flows:

We will perform simulation on the proposed scheme RARRA along with ARF and REACT by varying contending flows on the following topology:

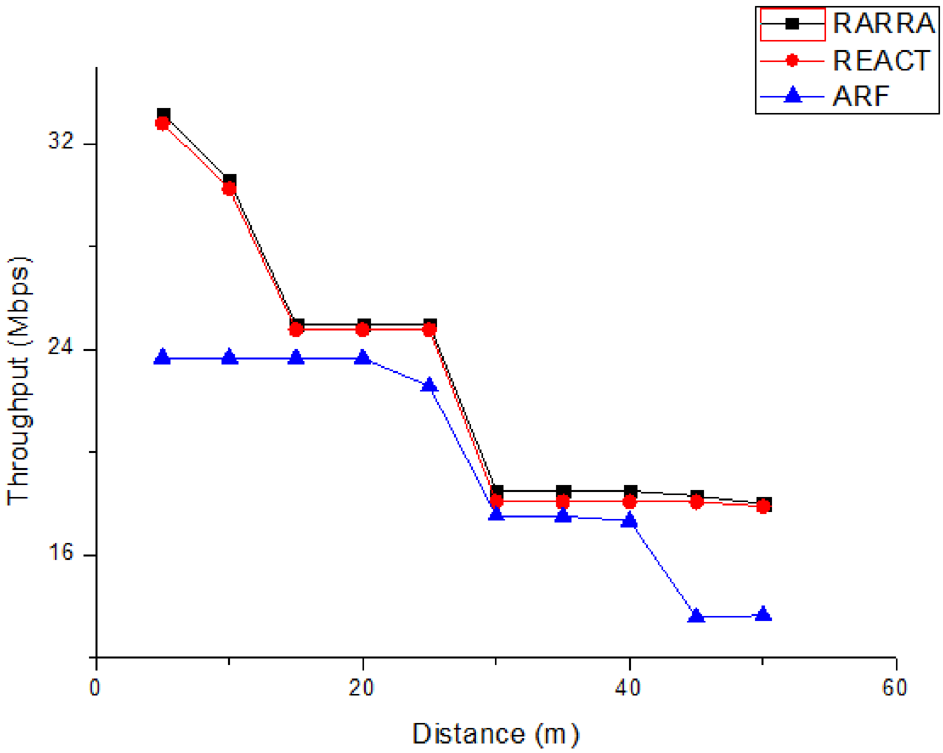
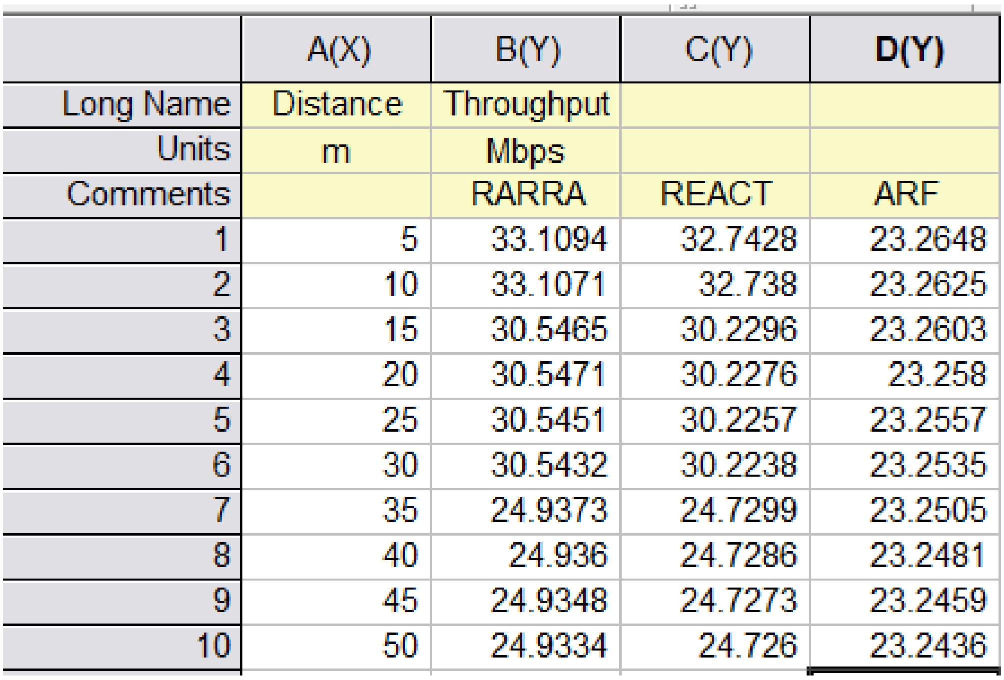


FIGURE 4.4: Graph showing the outcome for different algorithms in "Log distance path loss model" and "Randomwalk2d mobility model"

- **Type of network :** Infrastructure Based
- **Number of AP nodes :** 1
- **Number of Station nodes :** 1-10
- **Number of flows :** 1-10
- **Packet transmitted :** 100000
- **Packet size :** 2048 bytes
- **Flow data rate :** 54 Mbps
- **Path loss Model :** a) Log distance path loss model
  b) Two ray ground propagation loss model

- **AP location :** x=10 and y=25
- **Station nodes location :** In a grid of width = 2 , deltaX=20 , deltaY=5, bounded area=1000
- **AP node Mobility Model :** Constant position mobility model



	A(X)	B(Y)	C(Y)	D(Y)
Long Name	Distance	Throughput		
Units	m	Mbps		
Comments		RARRA	REACT	ARF
1	5	33.1094	32.7428	23.2648
2	10	33.1071	32.738	23.2625
3	15	30.5465	30.2296	23.2603
4	20	30.5471	30.2276	23.258
5	25	30.5451	30.2257	23.2557
6	30	30.5432	30.2238	23.2535
7	35	24.9373	24.7299	23.2505
8	40	24.936	24.7286	23.2481
9	45	24.9348	24.7273	23.2459
10	50	24.9334	24.726	23.2436

FIGURE 4.5: Table showing the results for different algorithms in "Two ray ground propagation loss model" and "Constant position mobility model"

- **Station nodes**
  Mobility Model : Randomwalk2d mobility model (2-4 uniform random speed)

Now we evaluate the results of RARRA, REACT, ARF in "Log distance path loss model" and "Two ray ground propagation loss model".

4.3.1 With various Contending Flows in "Log distance path loss model":

RESULT TABLE: The result table is represented in figure: 4.9

COMPARISON GRAPH: The comparison graph is represented in figure 4.10

4.3.2 With various Contending Flows in "Two ray ground propagation loss model":

RESULT TABLE: The result table is represented in figure: 4.11

COMPARISON GRAPH: The comparison graph is represented in figure 4.12

From the above data that we collected, it is apparent that both REACT[5] and RARRA follow the same trend in terms of throughput changes. This is mainly due

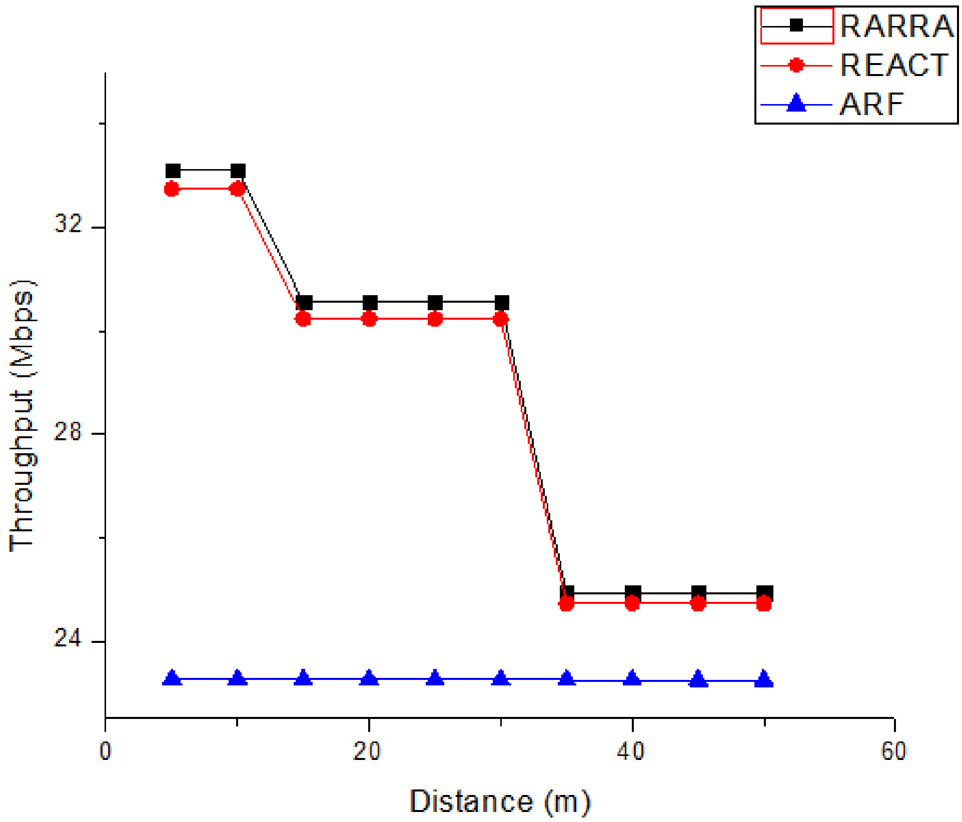


FIGURE 4.6: Graph showing the outcome for different algorithms in "Two ray ground propagation loss model" and "Constant position mobility model"

to the fact that we have used the same SNR range mapping in both algorithms to keep the conditions constant and vary only those parameters based on which comparisons are made. Here we have varied the number of contending flows with a static Access Point.

However, it is worth mentioning that our algorithm demonstrates a performance improvement in case of both path loss models. We have used LogDistancePathLoss model and TwoRayGroundPropagationLoss model for mobile node scenarios.

For LogDistancePathLoss model, the changes are less apparent, but still outperform REACT with increasing number of flows. But in case of TwoRayGround-PropagationLoss model when nodes are mobile, channel condition varies much and it is in this case that our algorithm shows more robustness as compared to the former path loss model. Here with increasing number of flows RARRA far outperforms REACT in cases when nodes are mobile and channel conditions are changing.

The above figures for each of the path loss models clearly demonstrate that RARRA performs much better than REACT in mobile scenarios when we have a static Access Point and the nodes around it are mobile. However, ARF[1] is

	A(X)	B(Y)	C(Y)	D(Y)
Long Name	Distance	Throughput		
Units	m	Mbps		
Comments		RARRA	REACT	ARF
1	5	33.1474	32.783	23.6714
2	10	33.1451	32.7808	23.669
3	15	30.5528	30.2438	23.6604
4	20	30.5507	30.2419	23.6571
5	25	30.5487	30.24	23.6547
6	30	30.5467	30.2481	23.6524
7	35	24.9598	24.7596	23.6561
8	40	24.9584	24.7583	23.6537
9	45	24.9571	24.757	23.6517
10	50	24.9558	24.7557	23.649

FIGURE 4.7: Table showing the results for different algorithms in "Two ray ground propagation loss model" and "Randomwalk2d mobility model"

always at the bottom of the list when each path loss model is used.

4.4 Results with various Packet sizes:

We will perform simulation on the proposed scheme RARRA along with ARF and REACT by varying packet sizes on the following topology:

- **Type of network :** Adhoc
- **Number of nodes :** 2
- **Number of flows :** 1
- **Packet transmitted :** 100000
- **Packet sizes :** 250 , 500 , 1000 , 2000 , 2048 bytes
- **Flow data rate :** 54 Mbps
- **Path loss Model :** a) Log distance path loss model b) Two ray ground propagation loss model
- **Mobility Model :** Constant position mobility model
- **Distance between nodes:** 30

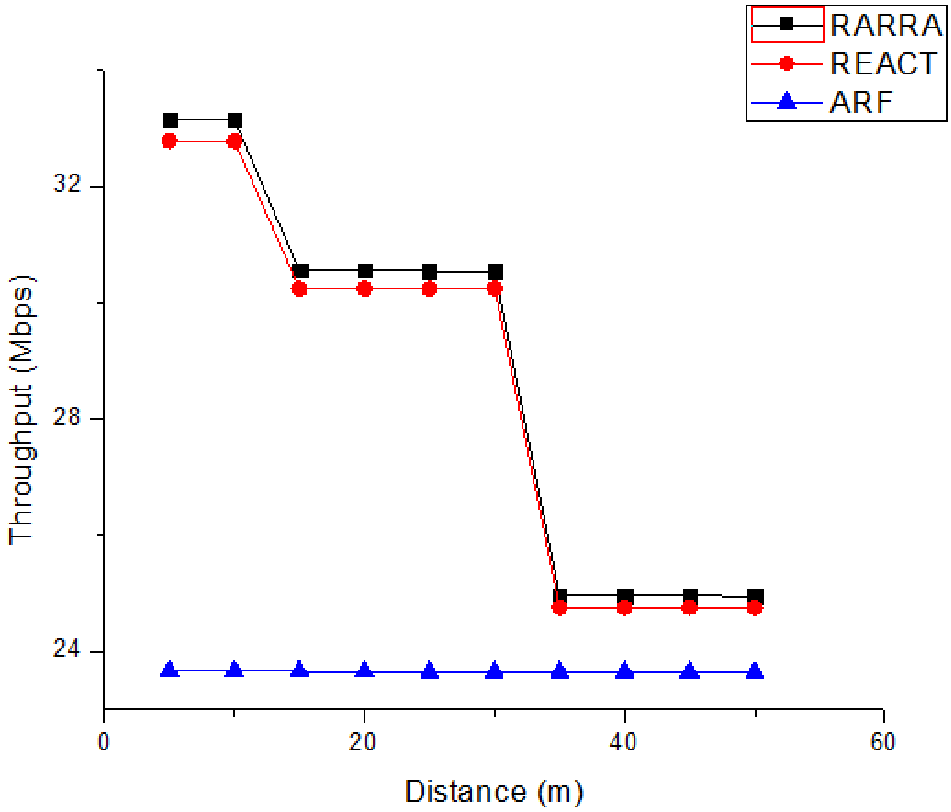


FIGURE 4.8: Graph showing the outcome for different algorithms in "Two ray ground propagation loss model" and "Randomwalk2d mobility model"

Now we evaluate the results of RARRA, REACT, ARF in "Log distance path loss model" and "Two ray ground propagation loss model".

4.4.1 With various Packet sizes in "Log distance path loss model":

RESULT TABLE: The result table is represented in figure: 4.13

COMPARISON GRAPH: The comparison graph is represented in figure 4.14

4.4.2 With various Packet sizes in "Two ray ground propagation loss model":

RESULT TABLE: The result table is represented in figure: 4.15

COMPARISON GRAPH: The comparison graph is represented in figure 4.16

From the above data that we collected, it is apparent that both REACT[5] and RARRA follow the same trend in terms of throughput changes. This is mainly due to the fact that we have used the same SNR range mapping in both algorithms

	A(X)	B(Y)	C(Y)	D(Y)
Long Name	No. of flows	Throughput		
Units		Mbps		
Comments		RARRA	REACT	ARF
1	1	24.9432	24.771	22.7814
2	2	12.8456	12.8662	10.9217
3	3	7.6148	7.8053	6.7446
4	4	5.4342	5.144	4.1607
5	5	4.1292	3.7256	2.1539
6	6	3.3624	2.8196	2.4657
7	7	2.676	2.3271	1.9699
8	8	2.3544	1.9622	1.5832
9	9	1.8024	1.7907	1.3405
10	10	1.7656	1.3522	0.8668

FIGURE 4.9: Table showing the results for different algorithms in "Log distance path loss model"

to keep the conditions constant and vary only those parameters based on which comparisons are made. Here we have varied the size of transmitted packets.
However, it is worth mentioning that our algorithm demonstrates a performance improvement in case of both path loss models. We have used LogDistancePathLoss model and TwoRayGroundPropagationLoss model for static node scenarios.
For TwoRayGroundPropagationLoss model, the changes are less apparent, but still outperform REACT with increasing size of transmitted packets. But in case of LogDistancePathLoss model where nodes are static, which is also the case for the former path loss model, our algorithm shows more robustness. Here with increasing size of transmitted packets RARRA far outperforms REACT in cases when nodes are static.
The above figures for each of the path loss models clearly demonstrate that RARRA performs much better than REACT in static scenarios when nodes are static when packet size is increasing mainly due to the instant feedback about channel conditions in case of RARRA and the optimal rate switching. However, ARF[1] is always at the bottom of the list when each path loss model is used.

4.5 Results with various Speeds:

We will perform simulation on the proposed scheme RARRA along with ARF and REACT by varying speeds on the following topology:

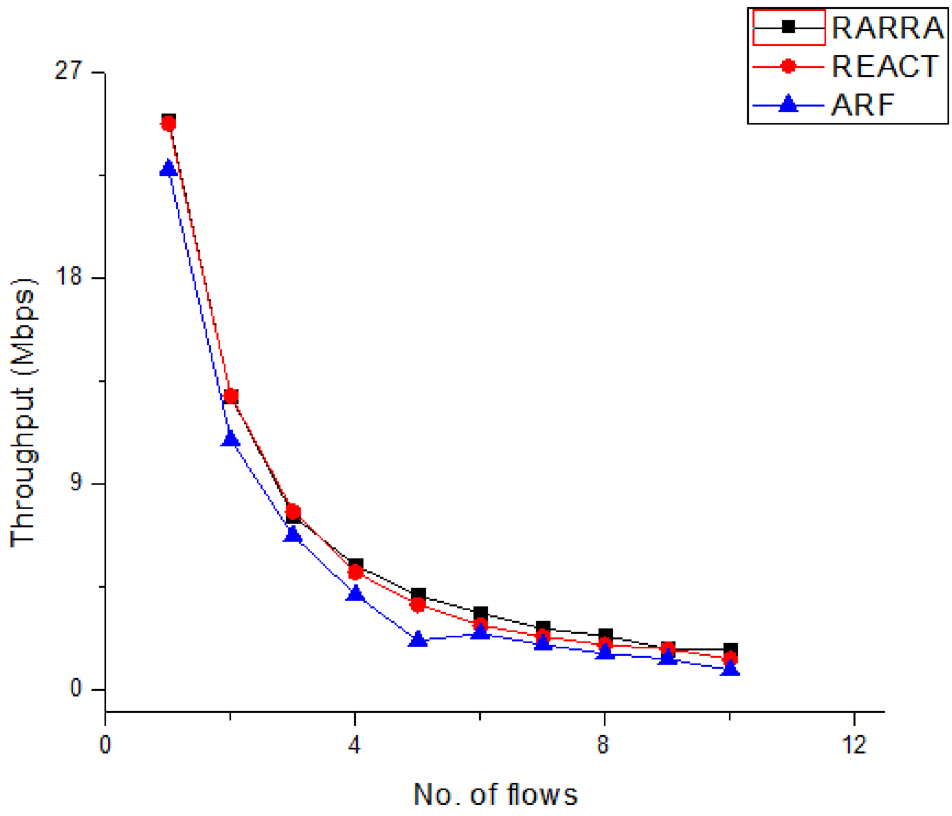


FIGURE 4.10: Graph showing the outcome for different algorithms in "Log distance path loss model"

- **Type of network :** Adhoc
- **Number of nodes :** 2
- **Number of flows :** 1
- **Packet transmitted :** 100000
- **Flow data rate :** 54 Mbps
- **Path loss Model :** a) Log distance path loss model b) Two ray ground propagation loss model
- **Mobility Model :** Randomwalk2d mobility model
- **Mobility speed :** speed between 0-2 , 2-4 , 4-6, 6-8 , 8-10 , 10-12 , 12-14 , 14-16 , 16-18, 18-20 (m/s).
- **Distance between nodes:** 30

Now we evaluate the results of RARRA, REACT, ARF in "Log distance path loss model" and "Two ray ground propagation loss model".

	A(X)	B(Y)	C(Y)	D(Y)
Long Name	No. of flows	Throughput		
Units		Mbps		
Comments		RARRA	REACT	ARF
1	1	30.5483	30.3005	22.8136
2	2	15.9696	15.8444	11.1178
3	3	9.3258	9.3151	6.869
4	4	6.5701	5.6626	3.4343
5	5	4.8281	4.1747	2.2928
6	6	3.9099	3.2471	2.4645
7	7	3.0949	2.348	1.9759
8	8	2.6504	1.896	1.3923
9	9	2.276	1.7341	1.2347
10	10	2.0299	1.6229	0.8066

Figure 4.11: Table showing the results for different algorithms in "Two ray ground propagation loss model"

4.5.1 With various Speeds in "Log distance path loss model":

RESULT TABLE: The result table is represented in figure: 4.17

COMPARISON GRAPH: The comparison graph is represented in figure 4.18

4.5.2 With various Speeds in "Two ray ground propagation loss model":

RESULT TABLE: The result table is represented in figure: 4.19

COMPARISON GRAPH: The comparison graph is represented in figure 4.20

From the above data that we collected, it is apparent that both REACT[5] and RARRA follow the same trend in terms of throughput changes. This is mainly due to the fact that we have used the same SNR range mapping in both algorithms to keep the conditions constant and vary only those parameters based on which comparisons are made. Here we have varied the speed of mobility of the nodes.
However, it is worth mentioning that our algorithm demonstrates a performance improvement in case of both path loss models. We have used LogDistancePathLoss model and TwoRayGroundPropagationLoss model for mobile node scenarios.
For LogDistancePathLoss model, the changes are less apparent, but still outperform REACT with increasing speed of mobile nodes. But in case of TwoRayGroundPropagationLoss model where nodes are mobile, which is also the case for

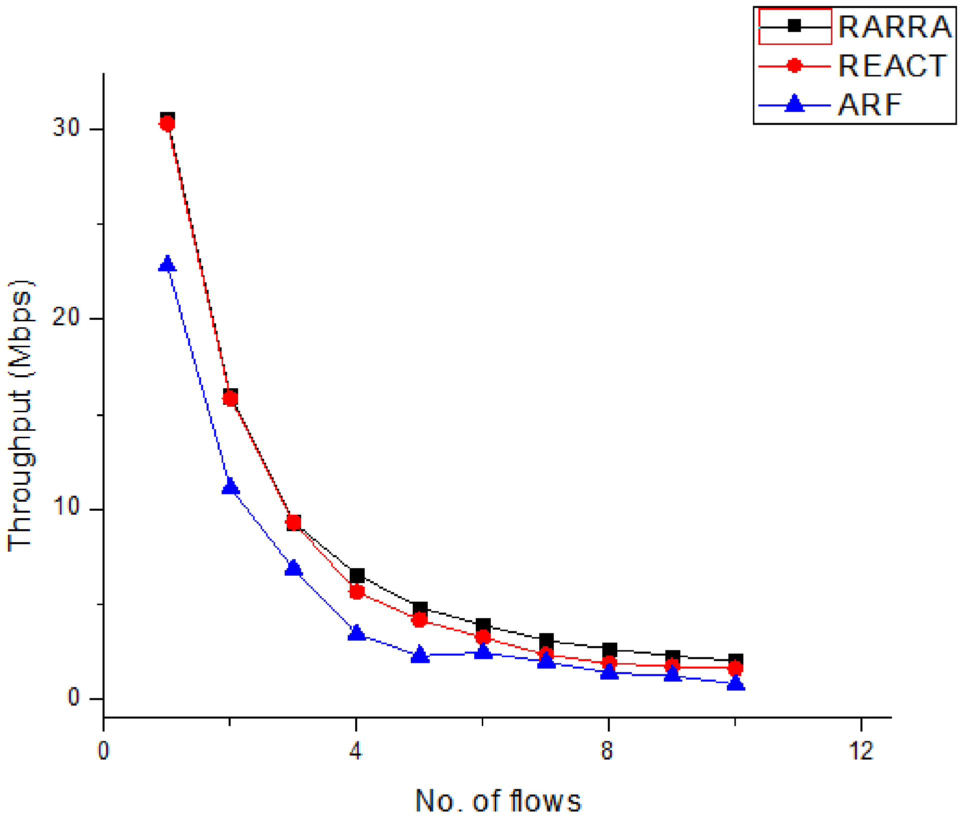


FIGURE 4.12: Graph showing the outcome for different algorithms in "Two ray ground propagation loss model"

the former path loss model, our algorithm shows more robustness. Here with increasing speed of mobile nodes RARRA far outperforms REACT in cases when nodes are mobile.

The above figures for each of the path loss models clearly demonstrate that RARRA performs much better than REACT in mobile scenarios when nodes are mobile, when speed of mobile nodes is increasing and channel condition is changing dynamically. This is mainly due to the instant feedback about channel conditions in case of RARRA as well as the optimal rate switching. However, ARF[1] is always at the bottom of the list when each path loss model is used.

	A(X)	B(Y)	C(Y)	D(Y)
Long Name	Packet Size	Throughput		
Units	Bytes	Mbps		
Comments		RARRA	REACT	ARF
1	250	7.76963	7.76985	7.0624
2	500	11.7225	11.2694	7.71063
3	1000	15.3032	15.024	11.76
4	1500	16.9745	16.8596	14.2359
5	2000	18.0093	17.9716	16.4095
6	2048	18.3093	18.0575	16.5452

FIGURE 4.13: Table showing the results for different algorithms in "Log distance path loss model"

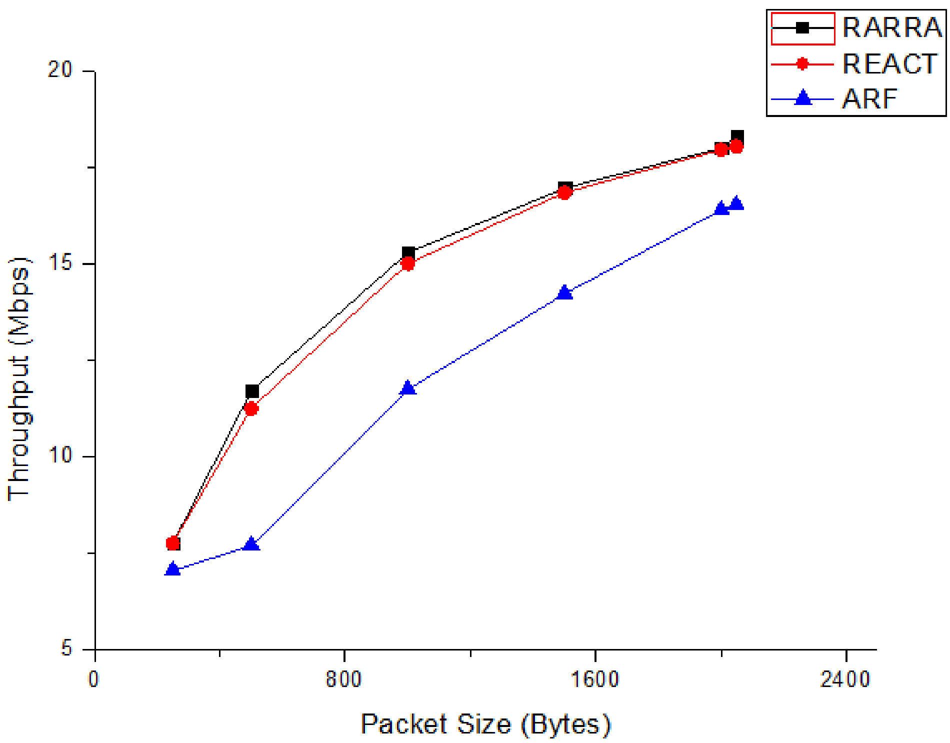


FIGURE 4.14: Graph showing the outcome for different algorithms in "Log distance path loss model"

	A(X)	B(Y)	C(Y)	**D(Y)**
Long Name	Packet Size	Throughput		
Units	bytes	Mbps		
Comments		RARRA	REACT	ARF
1	200	9.79673	9.61281	9.16867
2	500	15.677	15.4196	10.0251
3	1000	22.9531	22.6583	16.0326
4	1500	27.366	27.0499	20.0876
5	2000	30.3019	29.9886	23.0542
6	2048	30.5432	30.2238	23.2535

FIGURE 4.15: Table showing the results for different algorithms in "Two ray ground propagation loss model"

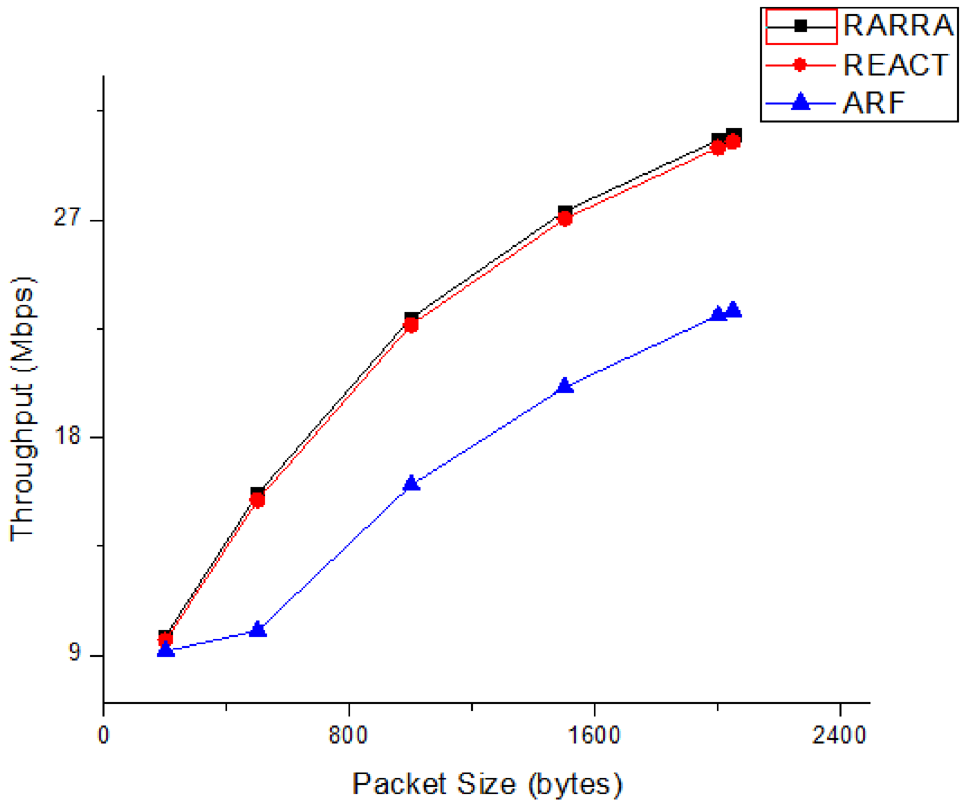


FIGURE 4.16: Graph showing the outcome for different algorithms in "Two ray ground propagation loss model"

	A(X)	B(Y)	C(Y)	D(Y)
Long Name	Speed	Throughput		
Units	m/s	Mbps		
Comments		RARRA	REACT	ARF
1	0-2	18.0925	18.073	18.5416
2	2-4	18.0925	18.0729	17.5341
3	4-6	18.0924	18.0728	17.4008
4	6-8	20.0997	21.2128	18.5903
5	8-10	24.3876	24.234	19.8391
6	10-12	24.9619	24.7617	21.1164
7	12-14	24.9624	24.7622	22.8882
8	14-16	24.9629	24.7626	23.6438
9	16-18	24.9631	24.7629	23.656
10	18-20	24.9629	24.7627	23.626

FIGURE 4.17: Table showing the results for different algorithms in "Log distance path loss model"

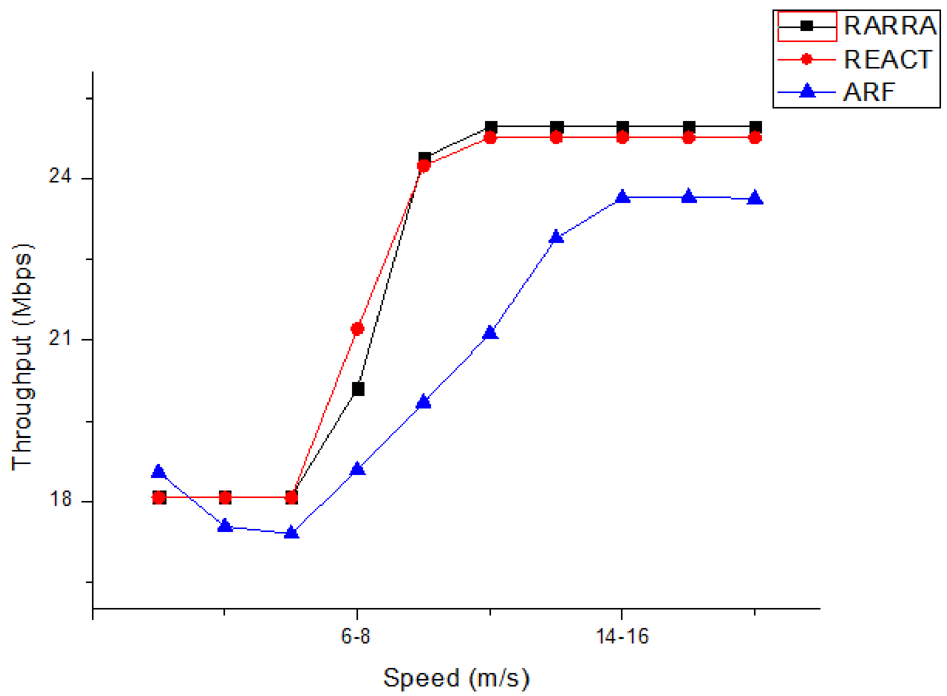


FIGURE 4.18: Graph showing the outcome for different algorithms in "Log distance path loss model"

	A(X)	B(Y)	C(Y)	D(Y)
Long Name	Speed	Throughput		
Units	m/s	Mbps		
Comments		RARRA	REACT	ARF
1	0-2	30.5469	30.2382	23.6526
2	2-4	30.5467	30.2481	23.6524
3	4-6	30.5464	30.2478	23.652
4	6-8	30.547	30.2383	23.6527
5	8-10	30.5475	30.2389	23.6534
6	10-12	30.548	30.2393	23.6538
7	12-14	30.5487	30.24	23.6549
8	14-16	30.5494	30.2407	23.6557
9	16-18	30.5498	30.2411	23.656
10	18-20	30.5495	30.2408	23.6557

FIGURE 4.19: Table showing the results for different algorithms in "Two ray ground propagation loss model"

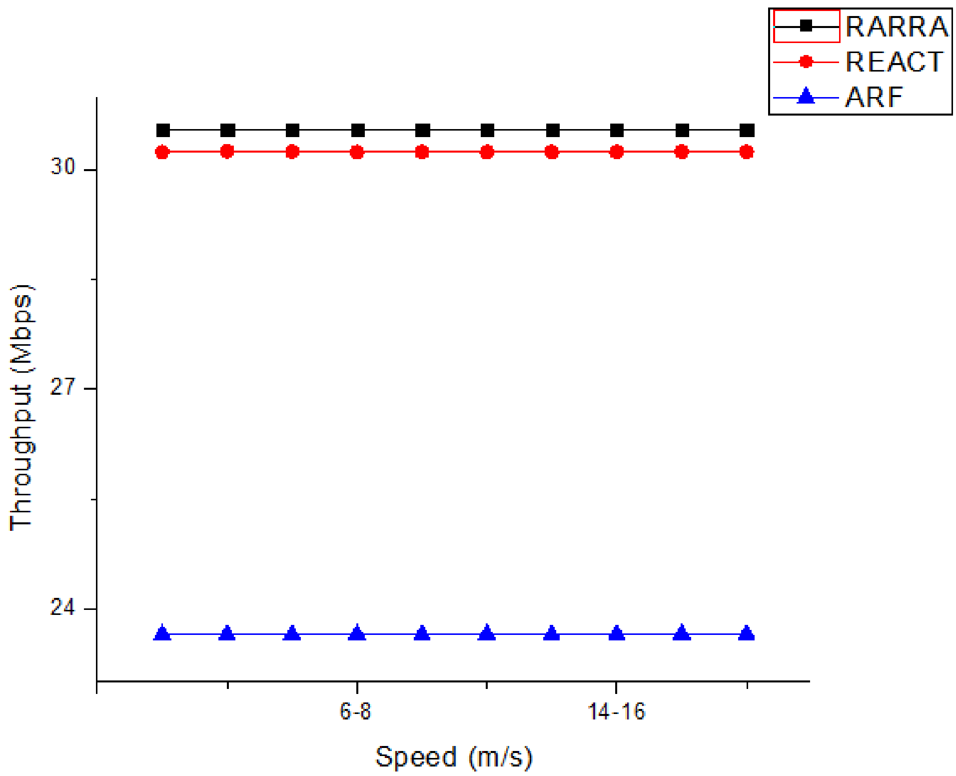


FIGURE 4.20: Graph showing the outcome for different algorithms in "Two ray ground propagation loss model"

Chapter 5

Conclusion

5.1 Summary of Contributions

Though much work has been done on the Rate Adaptation Techniques none of them meets all criteria for a robust and optimal method .So to provide all in one package ,we proposed a robust rate adaptation scheme in which Receiver controls the ACK transmission rate as a means to dictate the sender to adjust data rate. RARRA is also responsive to time-varying wireless channel owing to the accurate and instant feedback. Further, it mitigates rate avalanche effect through Adaptive RTS. Finally it can switch to optimal transmission rate. The advantages of RARRA over REACT[5] in terms of varying distances between both static and mobile nodes, speed of mobile nodes, varying packet sizes for static nodes and number of contending flows to a static Access Point is notable.

5.2 Future Work

Furthermore, we can see clearly from our simulation results that RARRA outperforms REACT especially in dynamic cases when nodes are mobile and channel conditions are dynamic. This robustness of our algorithm further makes it possible to be used in networks such as VANETs.

Bibliography

[1] M.Lacage,M.Manshaei, and T.Turletti, "IEEE 802.11 RateAdaptation: A Practical Approach", *in MSWiM04*, 2004, pp. 126- 134.

[2] J.C. Bicket, "Bit-rate Selection in Wireless Networks", *Master's thesis, Massachussettes Institute of Technology* , 2005.

[3] G.Holland , N.Vaidya, and P.Bahl, "A rate-adaptive MAC protocol for multihop wireless networks ", *in ACM MOBICOM'01*, 2001.

[4] G. Judd, X. Wang, and P. Steenkiste, "Efficient Channel-aware Rate Adaptation in Dynamic Environments ", *In Proceeding of the ACM MobiSys*, 2008.

[5] H.Jung, T.Kwon, K.Cho, and Y.Choi, "REACT: Rate Adaptation using Coherence Time in 802.11 WLANs", *Elsevier*, 2011.

[6] D. Qiao, S. Choi, and K.G. Shin, "Goodput analysis and link adaptation for IEEE 802.11a wireless lans", *IEEE Transactions on Mobile Computing 1 (4)* , 2002, pp. 278-292.

[7] T.S. Rappaport, "Wireless Communications: Principles and Practice", *PrenticeHall, NJ*, 2002.

[8] M. Carroll and T. Wysocki, "Fading characteristics for indoor wireless channels at5 GHz unlicensed bands", 2003, pp. 102-105.

[9] Pralhad Deshpande and Samir Das, "BRAVE: Bit-Rate Adaptation in Vehicular Environments", *in Proceedings of ACM VANET'12*, 2012.

[10] J. Camp and E. Knightly, "Modulation Rate Adaptation in Urban and Vehicular Environments: Cross-layer Implementation and Experimental Evaluation", *IEEE/ACM Transactions on Networking, 18(6):1949-1962*, December, 2010.