MASTER OF SCIENCE IN COMPUTER SCIENCE
AND ENGINEERING

# Receiver Assisted Rate Adaptation in Wireless Networks

by

**Nafiul Rashid**

**Department of Computer Science and Engineering (CSE)**
**Islamic University of Technology (IUT)**
**Organisation of Islamic Cooperation (OIC)**
**Gazipur-1704, Bangladesh**

**November, 2015**

# Receiver Assisted Rate Adaptation in Wireless Networks

**by**

**Nafiul Rashid**

Student No. 134604

**Supervised by**

**Prof. Dr. Muhammad Mahbub Alam**

A thesis submitted to the Department of CSE
in partial fulfillment of the requirements for the award of the degree of
Master of Science in Computer Science and Engineering
(M.Sc. Engg. in CSE)

**Department of Computer Science and Engineering (CSE)**
**Islamic University of Technology (IUT)**
**Organisation of Islamic Cooperation (OIC)**
**Gazipur-1704, Bangladesh**

**November, 2015**

# *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 NARC (Neighbor Aware Rate Control). Firstly, 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. We use acknowledgment rates to serve this feedback purpose. Our feedback mechanism also allows for optimal rate switch rather than sequential one that most of the existing methods support. Secondly, we address the stale feedback problem that the SNR based methods mainly suffer from and provide a unique solution to overcome this. To the best of our knowledge no works have addressed the solution to this problem. The stale feedback problem was mitigated by a prediction mechanism using linear regression on the observed rates on sender side and feedback rates from the receiver side. Besides, 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. NARC 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 sender side prediction mechanism to tackle against stale feedback problem. Moreover use of Adaptive RTS provides robustness to our method.

# Acknowledgment

At the very beginning I express my heartiest gratitude to Almighty Allah for His divine blessings which has allowed me to bring this research work to life.

I am ever grateful and indebted to my supervisor Dr. Muhammad Mahbub Alam, Professor, Department of Computer Science and Engineering, IUT. His supervision, knowledge and relentless support have time and again proved to be invaluable and allowed me to complete this endeavor successfully. Thanking him would be an understatement.

It was my pleasure to get the cooperation and coordination from Professor Dr. M.A. Mottalib, Head of CSE Department, IUT during various phases of the work. I am grateful to him for his constant and energetic guidance and valuable advice.

My deep appreciation extends to all the respected jury member of my thesis committee for their insightful comments and constructive criticism of my research work. Surely they have helped me to improve this research work greatly.

Lastly I would like to thank the faculty members of CSE Department, IUT who have helped to make my working environment a pleasant one, by providing a helpful set of eyes and ears when problems arose.

Finally, I take this opportunity to express the profound gratitude from my deep heart to my beloved parents and my dearest elder brother for their love and continuous support both spiritually and materially.

# Contents


**Abstract** i

**Acknowledgment** ii

**List of Figures** vi

**List of Tables** vii

**1 Introduction** 1
- 1.1 Overview . . . 1
  - 1.1.1 What is rate adaptation? . . . 1
  - 1.1.2 What are the different standard rates? . . . 2
  - 1.1.3 Example of Rate Adaptation . . . 2
  - 1.1.4 Importance of Rate Adaptation . . . 3
  - 1.1.5 Classification of Rate Adaptation approaches . . . 3
  - 1.1.6 Different rate switching techniques . . . 4
  - 1.1.7 Hidden Node and Collision . . . 4
  - 1.1.8 Rate Avalanche Effect . . . 5
  - 1.1.9 Use of RTS/CTS . . . 6
- 1.2 Problem Statement . . . 7
- 1.3 Thesis Objectives . . . 9
- 1.4 Thesis Contributions . . . 9
- 1.5 Organization of The Thesis . . . 10

**2 Literature Review** 11
- 2.1 Related Work . . . 11
  - 2.1.1 Frame Based Methods . . . 11
    - 2.1.1.1 ARF(Auto Rate Fallback) . . . 11
    - 2.1.1.2 CARA(Collision Aware Rate Adaptation) . . . 12
    - 2.1.1.3 SampleRate . . . 12
    - 2.1.1.4 Ministrel . . . 12
    - 2.1.1.5 AMRR(Adaptive Multi Rate Retry) . . . 13
    - 2.1.1.6 RRAA(Robust Rate Adaptation Algorithm) . . . 13
    - 2.1.1.7 HERA(Hidden node Effect aware Rate Adaptation) . . . 14
  - 2.1.2 SNR Based Methods . . . 15
    - 2.1.2.1 RBAR(Receiver Based AutoRate) . . . 15
    - 2.1.2.2 REACT(Rate Adaptation Using Coherence Time) . . . 16
- 2.2 Critiques On Related Works . . . 19
  - 2.2.1 Frame Based or SNR Based . . . 19

2.2.2 Channel Quality Estimation Side . . . . . 19
2.2.3 Rate Switching Techniques used . . . . . 19
2.2.4 Use of RTS/CTS . . . . . 20
2.2.5 Differentiating the Cause of Frame Loss . . . . . 20

**3 Proposed Method-NARC** **22**
3.1 NARC-Receiver Side Mechanism . . . . . 22
3.1.1 Stepwise procedure . . . . . 22
3.1.2 Acknowledgment Rate mapping table . . . . . 23
3.1.3 Receiver Side Mechanism Flow Diagram . . . . . 23
3.2 NARC-Sender Side Mechanism . . . . . 24
3.2.1 Stepwise Procedure . . . . . 24
3.2.2 Demonstration . . . . . 25
3.2.3 Linear regression equations . . . . . 26
3.2.4 Initial Condition . . . . . 27
3.2.5 Transmission failure . . . . . 27

**4 Simulation Results and Performance Evaluation** **28**
4.1 Simulation Setup . . . . . 28
4.2 Performance Evaluation . . . . . 28
4.2.1 Varying distance in static scenario: . . . . . 29
4.2.1.1 Topology . . . . . 29
4.2.1.2 Result . . . . . 29
4.2.1.3 Analysis . . . . . 30
4.2.2 Varying distance in mobile scenario: . . . . . 31
4.2.2.1 Topology . . . . . 31
4.2.2.2 Result . . . . . 31
4.2.2.3 Analysis . . . . . 31
4.2.3 Hidden node scenario: . . . . . 32
4.2.3.1 Topology . . . . . 32
4.2.3.2 Result . . . . . 32
4.2.3.3 Analysis . . . . . 33
4.2.4 Multiple flows scenario: . . . . . 34
4.2.4.1 Topology . . . . . 34
4.2.4.2 Result . . . . . 34
4.2.4.3 Analysis . . . . . 34
4.2.5 Multiple flows at sender side: . . . . . 35
4.2.5.1 Topology . . . . . 35
4.2.5.2 Result . . . . . 36
4.2.5.3 Analysis . . . . . 36
4.2.6 Multiple flows at receiver side: . . . . . 37
4.2.6.1 Topology . . . . . 37
4.2.6.2 Result . . . . . 38
4.2.6.3 Analysis . . . . . 39
4.2.7 Varying number of contending flows: . . . . . 39
4.2.7.1 Topology . . . . . 39
4.2.7.2 Result . . . . . 40
4.2.7.3 Analysis . . . . . 40
4.2.8 Varying packet size: . . . . . 41

4.2.8.1 Topology . . . . . . . . . . . . . . . . . . . . . . . . . . 41
4.2.8.2 Result . . . . . . . . . . . . . . . . . . . . . . . . . . . . 42
4.2.8.3 Analysis . . . . . . . . . . . . . . . . . . . . . . . . . . 42

**5 Conclusion** **43**
5.1 Summary of Contributions . . . . . . . . . . . . . . . . . . . . . . . . . 43
5.2 Future Work . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 43

# List of Figures


1.1 Channel conditions in different scenario . . . 2
1.2 Hidden nodes and Collision Scenario . . . 5
1.3 Effect of Rate Avalanche . . . 6
1.4 RTS/CTS handshake to avoid hidden node collision . . . 7
1.5 Demostration of stale feedback problem . . . 8

2.1 Standard MAC Frame format in IEEE 802.11 . . . 16
2.2 MAC and Physical Layer Frame format in RBAR . . . 16
2.3 REACT opeartional mechanism . . . 18

3.1 SNR-DATA rate Lookup table . . . 22
3.2 Different Acknowledgment rate corresponding to next suitable data rate . 23
3.3 Receiver side mechanism flow diagram . . . 24
3.4 Demonstration of rate selection at sender side . . . 25

4.1 Toplogy of two static nodes . . . 29
4.2 Throughput for various distance in static scenario . . . 30
4.3 Toplogy of two mobile nodes . . . 31
4.4 Throughput for various distance in mobile scenario . . . 32
4.5 Hidden node topology . . . 33
4.6 Throughput for various distance in mobile scenario . . . 33
4.7 Multiple flows topology . . . 34
4.8 Throughput for multiple flow scenario . . . 35
4.9 Topology of multiple flows at sender side . . . 36
4.10 Throughput for multiple flows at sender side . . . 36
4.11 Flow throughput from $n_2$ to $n_3$ . . . 37
4.12 Topology of multiple flows at receiver side . . . 38
4.13 Throughput for multiple flows at receiver side . . . 38
4.14 Flow throughput from $n_2$ to $n_3$ . . . 39
4.15 Topology for various number of contending flows . . . 40
4.16 Throughput for various number of contending flows . . . 41
4.17 Topology of two nodes with different packet size . . . 42
4.18 Throughput for various packet size in static scenario . . . 42

5.1 Rate adaptation algorithms in a nutshell . . . 44

# List of Tables

1.1 Different WLAN standards and their supported rates . . . . . . . . . . . . . 2
1.2 Effect of very high and low rates . . . . . . . . . . . . . . . . . . . . . . . . 3

2.1 RRAA implementation parameters for 802.11a rates . . . . . . . . . . . . 14
2.2 FER threshold value for rate decrease decision . . . . . . . . . . . . . . . 15

# Chapter 1

# Introduction

## 1.1 Overview

The IEEE 802.11 wireless local area network (WLAN) standard [1] 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 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 of the fundamental problems 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?

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".
It is the road map 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?

An ideal 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 [2]:

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.

### 1.1.3 Example of Rate Adaptation

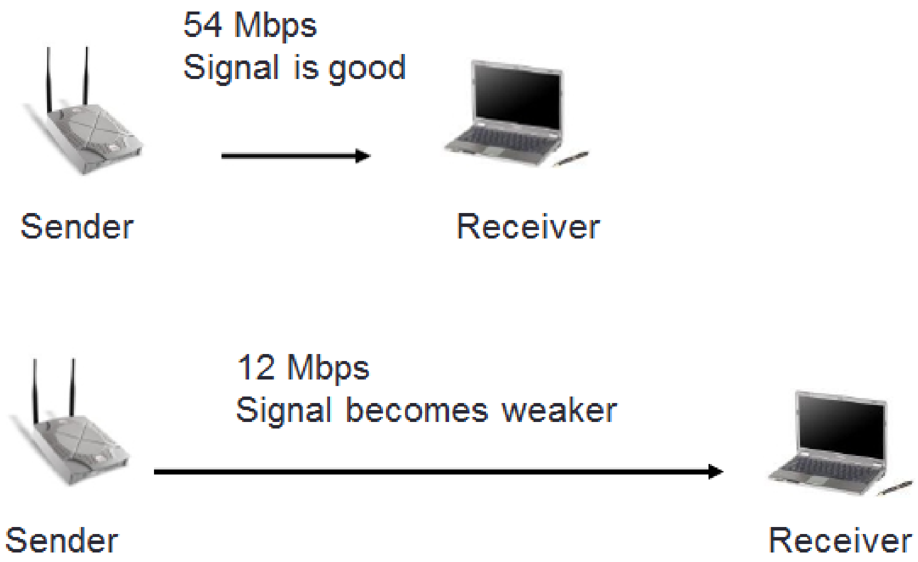



FIGURE 1.1: Channel conditions in different scenario

Here in the Fig. 1.1 the transmission rate should be adjusted according to the channel condition. 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 |

### 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 throughput 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:-

1 SNR based/Best RAs.

2 Frame based/Loss based RAs.

**SNR based RAs:**

- Uses physical layer metric i.e., SNR(Signal to Noise Ratio) [3] values to estimate channel quality.
- SNR-based designs translate the measured SNR into a transmission rate based on predefined mappings.

- 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 yields good channel utilization.

**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.
- 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 when 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.

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

### 1.1.6 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.

### 1.1.7 Hidden Node and Collision

Wireless networks use CSMA/CA technique to access the channel. CSMA/CA [4] is a channel access mechanism in shared media where multiple stations are contending to get the channel access.It follows a collision avoidance mechanism to access the channel. Still collision may take place due to hidden nodes [5]. Here Fig. 1.2 shows the hidden node scenario and how collision happens in that case. Three nodes A, B and C in the figure are placed. The oval shaped region represents the range of the corresponding node in the center. Thus node A is within the range of both B and C. However B and C are not within each others range. So when both B and C starts transmission at the same time they don't know the presence of other node as they are hidden from one another.Fig. 1.2 shows how a collision can take place due to hidden nodes.

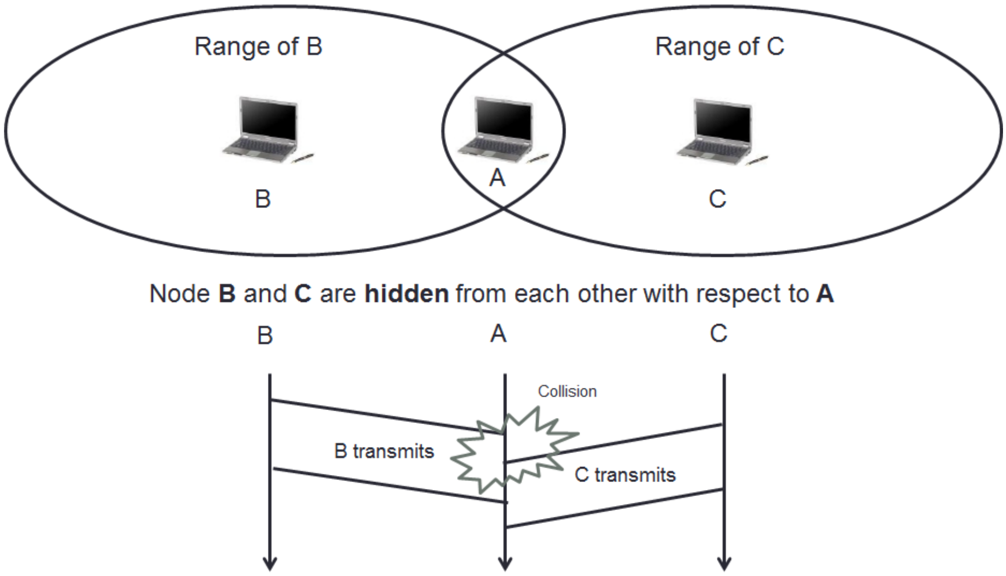



FIGURE 1.2: Hidden nodes and Collision Scenario

### 1.1.8 Rate Avalanche Effect

Here in the Fig. 1.3 it discusses a very important issue i.e., the rate avalanche effect [6]. 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.

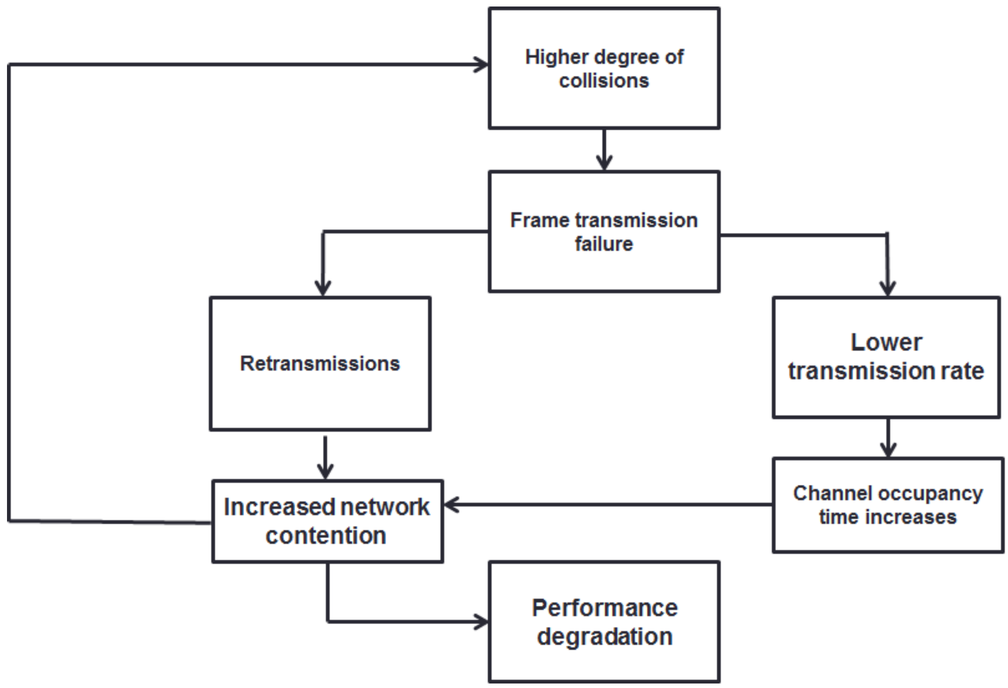



FIGURE 1.3: Effect of Rate Avalanche

- 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.

### 1.1.9 Use of RTS/CTS

RTS (Request To Send) and CTS (Clear To Send) , these two control frames are used to avoid collisions in a hidden node scenario [5].
**Pros:**

- Used to prevent collision from a hidden station via handshake
- Help to differentiate the cause of frame loss
- Thus avoid rate avalanche effect

**Cons:** These control frames incurs a lot of overhead in the network throughput. As they don't contribute to the overall throughput of the network their excessive use can degrade the network throughput. Fig 1.4 shows how RTS/CTS handshake can help to avoid the hidden node collision. At first node B sends an RTS frame to A. Upon receiving the RTS frame node A broadcasts a CTS frame to all its neighbors. Thus C gets to know that B is willing to communicate with A thereby refraining from transmission to A.

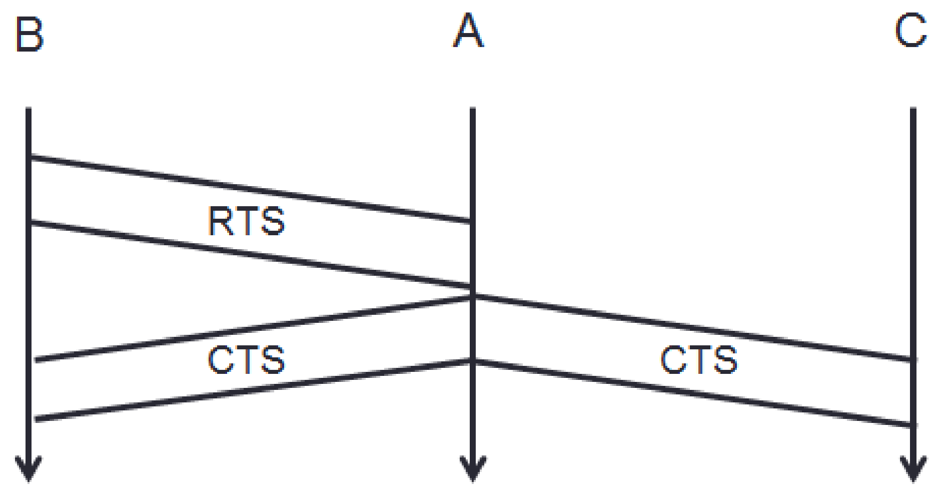



FIGURE 1.4: RTS/CTS handshake to avoid hidden node collision

## 1.2 Problem Statement

So far from 1.1 we have learned the basic of rate adaptations. Now let us go through the problems and challenges that the state-of-the-art rate adaptation algorithms suffer from:-

- Almost all **Frame Based** methods [7] –[13] **assumes channel quality** based on previous success or failure instead of **precise calculation of channel quality** like the **SNR Based** method.
- In all the **Frame Based** methods the channel quality is assumed solely from the **sender side** where as channel quality is **best measured** at the **receiver**.
- **SNR Based** method **RBAR** [14] measures the channel quality at receiver and informs the sender via modified CTS frame.

  However it needs RTS/CTS before every transmission thus incurs a lot of overhead.

  Becomes infeasible for off the shelf devices as a result of modified CTS frame.
- **SNR Based** method **REACT** [15] also measures the channel quality at the receiver but informs the sender via acknowledgment rate.

  However it allows sequential rate increase or decrease only.

Therefore we need a rate adaptation mechanism that incorporates the following properties:

1 SNR based method

2 Measures channel quality at the receiver

3 Informs the sender with less overhead

4 Allows multiple rate increase/decrease

We could have end here. But before that let's consider a scenario in Fig. 1.5. There are 7 nodes placed in an arbitrary manner. Node **A** and **C** communicates with node **B**, node **D** with node **E** and node **F** communicates with node **G**. We present an example scenario of their data transmission. Suppose node **A** transmits data to node **B** at 18 Mbps. Upon receiving the packet, node **B** measures the SNR and finds the channel to be improved to support 24 Mbps. So it acknowledges node **A** to send the next packet at 24 Mbps. However node **A** does not get the channel access immediately as it is a shared media and node **D** gets the access. It sends the data at 12 Mbps to **E** and gets the feedback to send the next packet at 12 Mbps. After that node **F** gets the access and transmits data at 12 Mbps to **G**. Node **G** similarly measures the channel to support 12 Mbps and acknowledge for same rate again. Next node **C** gets the access and transmits data to **B** at 18 Mbps and gets the feedback for 18 Mbps as well. That means now B is supporting 18 Mbps rate. Now if **A** gets the access immediately after **C** he will transmit data to **B** based on his last feedback that is 24 Mbps. However, **B** now supports 18 Mbps that means **A** is transmitting based on stale feedback. And this stale feedback problem will happen for every other nodes as well.

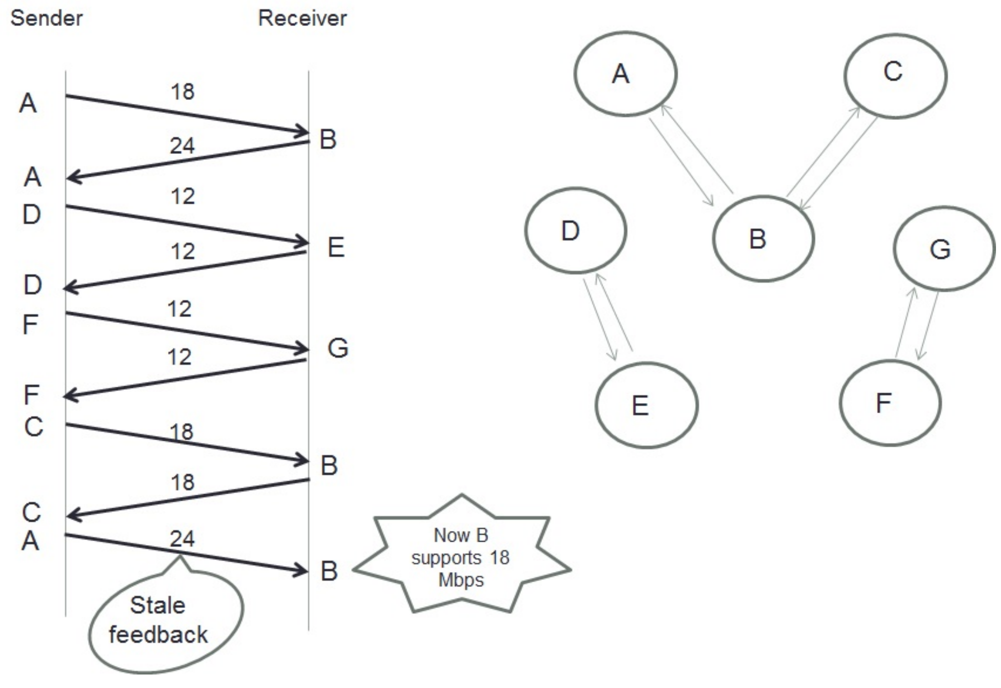



FIGURE 1.5: Demostration of stale feedback problem

So designing a SNR based rate adaptation method and sending the channel condition information from receiver to sender with less overhead does not solve it all, it has to tackle against this stale feedback problem. And to the best of our knowledge no algorithm so far has addressed the solution to this problem. That's why no SNR based method has been popular or commercially deployed so far. Therefore to establish the SNR based method suitable for commercial use one has to address the solution to the stale feedback problem as well.

## 1.3 Thesis Objectives

As mentioned in 1.2, each of existing algorithm meets specific criteria but does not fulfill all the criteria that determines an algorithm to be robust and optimal to highly dynamic channel condition. 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 an 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.
- **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 rather than sequential approach.
- **Overcome the stale feedback problem :**As this the key problem that hinders SNR based algorithms' performance.

## 1.4 Thesis Contributions

Keeping in mind the objectives stated in 1.3, we developed a Neighbor Aware Rate Control (NARC) algorithm. The contributions of our method are two fold:-

1 . **Receiver Side Mechanism :** The receiver side mechanism of our algorithm allows the receiver to inform the more precise channel condition to the sender. The feedback mechanism uses acknowledgment rate like REACT and incurs no overhead avoiding the use of RTS/CTS before every transmission like RBAR. However unlike REACT our algorithm allows the sender for multiple rate increase or decrease.

2 . **Sender Side Mechanism :** The sender side mechanism of our algorithm deals with the stale feedback problem. We developed a method where the sender uses its neighborhood information as well as the receiver's feedback to predict the next transmission rate using linear regression method. Thus it avoids the data transmission based on stale feedback and cope up with the highly dynamic channel condition.

## 1.5 Organization of The Thesis

The rest of the thesis will be organized as follows:in Chapter 2, we present the literature review of existing methods as well as their limitations in terms of rate adaptation parameters. In Chapter 3, we give a detailed description of our proposed method NARC and necessary background study. There, we discuss about the overall idea of our proposed algorithm and stepwise procedure. In Chapter 4, simulation set up, simulation results and performance analysis of our proposed method along with various existing ones are discussed.Finally, in Chapter 5, we conclude with the summary of our thesis contributions and provide the future directions of further developing our work.

# 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 Frame Based Methods

#### 2.1.1.1 ARF(Auto Rate Fallback)

ARF [7] 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.

#### 2.1.1.2 CARA(Collision Aware Rate Adaptation)

CARA [8] was developed to overcome the limitations of ARF that is to differentiate the cause of frame loss whether it is due to channel error or due to hidden node collision. To this end they added the mechanism of using RTS frame every time a packet gets failed. They used the RTS frame before the transmission following a failure. Otherwise the rate increase or decrease mechanism of CARA is similar to that of ARF. Although they used RTS adaptively sometimes it suffers from frequent RTS on and off.

#### 2.1.1.3 SampleRate

SampleRate [9] 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 every tenth successful transmission it randomly choose from the higher rates.

#### 2.1.1.4 Ministrel

Ministrel [10] is another algorithm that is an improved version of SampleRate. It is a popular rate control algorithm in Linux based platform. It is also incorporated in the commercial devices like Mad-Wifi drivers.
The algorithm works as follows:

- It calculates an exponential weighted moving average of throughput after every transmission.
- If the current throughput is higher than the previous one it randomly chooses for available higher rates and transmits the next packet at that rate.
- If the transmission at the new higher rate gets failed it decrease to the previous rate with highest throughput.
- Otherwise on consecutive 4 transmission failure it decreases to lower rate.

In brief, Ministrel being an improved version of SampleRate it considers exponential weighted moving average of throughput as a parameter to increase to random higher rate instead of trying on every tenth successful transmission like SampleRate. Other methods are same as SampleRate. It is to mention that both SampleRate and Ministrel never use RTS for their transmission that is do not differentiate the cause of frame loss.

#### 2.1.1.5 AMRR(Adaptive Multi Rate Retry)

AMRR [11] is another frame based algorithm that follows sequential approach of switching rates. Unlike other frame based method it uses the statistics of short time period to make the rate increase or decrease decisions.
The algorithm works as follows:

- It increases to the next higher rate if in each of its last 10 successive 500 ms time slots at least 10 packets were transmitted and less than 10% of packets transmission failed.
- It decrease to the next lower rate if more than 33% of packets fail in 500 ms time slot.

Unlike other farme based methods AMRR uses time period statistics to make rate switch however it also avoids RTS use to differentiate the cause of frame loss.

#### 2.1.1.6 RRAA(Robust Rate Adaptation Algorithm)

RRAA [12] is also a frame based algorithm which uses three parameters associated with each rate to make the rate switching decisions sequentially. The three parameters are:

1 Estimation Window Size (ewnd)

2 Maximum Tolerable Loss threshold (MTL)

3 Opportunistic Rate Increase threshold (ORI)

The algorithm works as follows:

- All the frames in an estimation window are transmitted at the same rate.
- At the end of transmission of estimated window frames, a loss ratio is calculated for the transmitted frames in that window.
- If the loss ratio is greater than the MTL for the transmission rate of that window frames then it decrease to immediate lower rate.
- On the other hand if the loss ratio is less than the ORI of that transmitted rate then it increase to the next higher rate.
- It uses adaptive RTS in case of transmission failure. But it maintains an window of RTS.

The associated Estimation Window Size (ewnd), Maximum Tolerable Loss threshold (MTL) and Opportunistic Rate Increase threshold (ORI) for each of the 802.11a rates are shown in table 2.1

Table 2.1: RRAA implementation parameters for 802.11a rates

| **Rate**(Mbps) | **Critical Loss Ratio**(%) | **ORI** | **MTL** | ***ewnd*** |
|---|---|---|---|---|
| 6 | N/A | 50.00 | N/A | 6 |
| 9 | 31.45 | 14.34 | 39.32 | 10 |
| 12 | 22.94 | 18.61 | 28.68 | 20 |
| 18 | 29.78 | 13.25 | 37.22 | 20 |
| 24 | 21.20 | 16.81 | 26.50 | 40 |
| 36 | 26.90 | 11.50 | 33.63 | 40 |
| 48 | 18.40 | 4.70 | 23.00 | 40 |
| 54 | 7.52 | N/A | 9.40 | 40 |

#### 2.1.1.7 HERA(Hidden node Effect aware Rate Adaptation)

HERA [13] is another frame based algorithm that follows sequential approach of switching rates. Unlike other frame based method it is mainly designed for AP nodes. It is an improved version of AARF to deal with hidden node scenario. But this algorithm is not applied for general nodes. Here each rate is associated with error threshold.
The algorithm works as follows:

- It increases to the next higher rate on 10 consecutive success.

- It decrease to the next lower rate if the frame error rate of last 10 transmission is greater than the error threshold of the current rate.
- It also maintains an window of RTS frames to avoid the rate avalanche effect.

Table 2.2 shows the frame error threshold for each rate

Table 2.2: FER threshold value for rate decrease decision

| **Rate**(Mbps) | **Error Threshold** |
|---|---|
| 6 | N/A |
| 9 | 4 |
| 12 | 4 |
| 18 | 4 |
| 24 | 3 |
| 36 | 4 |
| 48 | 3 |
| 54 | 2 |

### 2.1.2 SNR Based Methods

#### 2.1.2.1 RBAR(Receiver Based AutoRate)

RBAR [14] 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 (Received Signal Strength Indicator) 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.

The Fig. 2.1 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 as shown in Fig. 2.2.

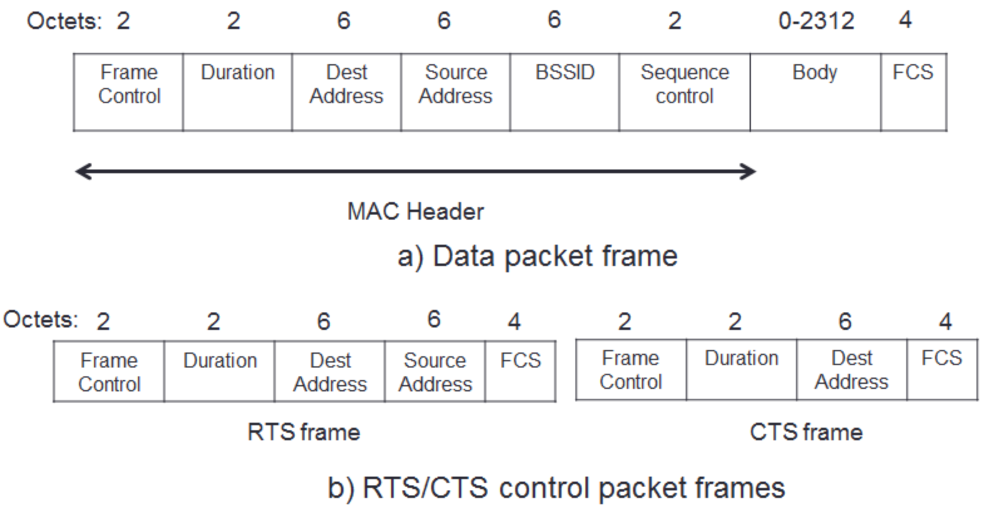



FIGURE 2.1: Standard MAC Frame format in IEEE 802.11

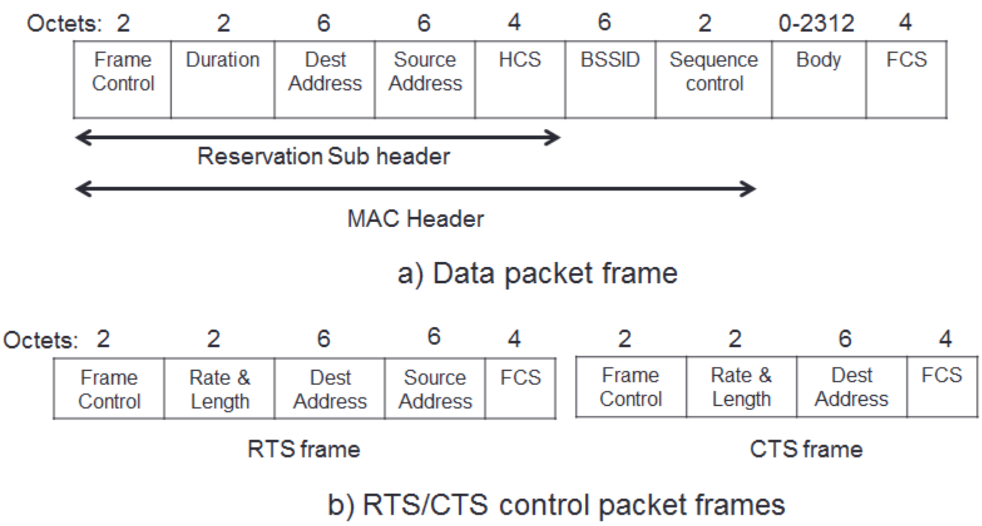



FIGURE 2.2: MAC and Physical Layer Frame format in RBAR

#### 2.1.2.2 REACT(Rate Adaptation Using Coherence Time)

REACT [15] is another SNR based approach whose key features are given below:

**Key features:-**

- The receiver in REACT informs the transmitter of the improved channel condition via altering the ACK(Acknowledgment) 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]:
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.

- 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.

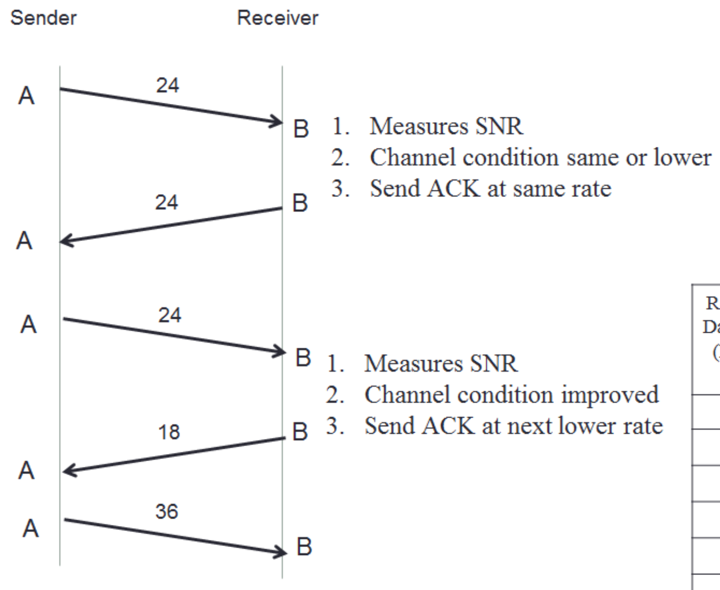
**When to decrease the data rate:-**

- 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.

The following Fig. 2.3 shows how REACT works :




| Received Data Rate (Mbps) | ACK Rate (Mbps) |
|---|---|
| 6 | 6 |
| 12 | 12 |
| 18 | 18 |
| 24 | 24 |
| 36 | 24 |
| 48 | 24 |
| 54 | 24 |

| Received Data Rate (Mbps) | Next Higher Rate (Mbps) | ACK Rate (Mbps) |
|---|---|---|
| 6 | 12 | 9 |
| 12 | 18 | 9 |
| 18 | 24 | 12 |
| 24 | 36 | 18 |
| 36 | 48 | 18 |
| 48 | 54 | 18 |

FIGURE 2.3: REACT opeartional mechanism

## 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, CARA, SampleRate, Ministrel, AMRR, RRAA and HERA are frame based approaches each of which uses success/failure of previously transmitted frames and switches rate sequentially or randomly. 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 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 Side

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, CARA, SampleRate, Ministrel, AMRR, RRAA and HERA measure channel condition at the sender 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.
Along with the frame based methods ARF, CARA, AMRR, RRAA and HERA the SNR based method REACT also 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 uses optimal rate switching techniques. It uses the rate advertised by the CTS frame. Hence the rate is calibrated with the channel conditions.
Lastly, SampleRate and Ministrel 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, Ministrel, AMRR never use RTS/CTS which reduces overhead but increases vulnerability of collision based losses.
CARA, RRAA, HERA, 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 them that uses RTS/CTS on demand. They incur the marginal overhead with the use of RTS/CTS.

### 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 select lower 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, Ministrel, AMRR 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, CARA, RRAA, HERA, REACT use Adaptive RTS. They use 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. An RTS window gives protection to only a few frames. So overhead is reduced but differentiation of cause is achieved.

# Chapter 3

# Proposed Method-NARC

## 3.1 NARC-Receiver Side Mechanism

### 3.1.1 Stepwise procedure

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

1 . We maintain a table like REACT that maps different ACK rate to different DATA rates.

2 . Determine the SNR of received DATA frame and select the next suitable DATA rate according to Fig. 3.1.

3 . For the next suitable DATA rate we look into our table select the corresponding ACK rate based on received DATA rate.

4 . Then we send the ACK(Acknowledgment) at the selected ACK rate.

The SNR-DATA rate lookup table is given below : This SNR-DATA rate lookup table

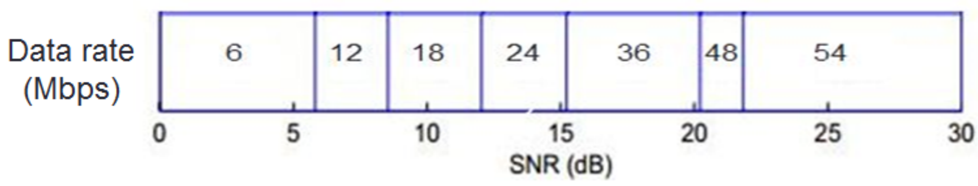



FIGURE 3.1: SNR-DATA rate Lookup table

in [16] 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.

### 3.1.2 Acknowledgment Rate mapping table

The corresponding acknowledgment rate based on the selected suitable data rate from SNR measurement is given in the following Fig. 3.2.

REACT

Same or Poor channel

| Received Data Rate (Mbps) | ACK Rate (Mbps) |
|---|---|
| 6 | 6 |
| 12 | 12 |
| 18 | 18 |
| 24 | 24 |
| 36 | 24 |
| 48 | 24 |
| 54 | 24 |

Improved channel

| Received Data Rate (Mbps) | Next Higher Rate (Mbps) | ACK Rate (Mbps) |
|---|---|---|
| 6 | 12 | 9 |
| 12 | 18 | 9 |
| 18 | 24 | 12 |
| 24 | 36 | 18 |
| 36 | 48 | 18 |
| 48 | 54 | 18 |

NARC

| Received Data Rate (Mbps) | Next Data Rate (Mbps) | ACK Rate (Mbps) |
|---|---|---|
| 6 | 6 | 6 |
| | | |
| 12 | 6 | 6 |
| | 12 | 9 |
| | 18 | 12 |
| | | |
| 18 | 12 | 6 |
| | 18 | 9 |
| | 24 | 12 |
| | 36 | 18 |

| Received Data Rate (Mbps) | Next Data Rate (Mbps) | ACK Rate (Mbps) |
|---|---|---|
| 24 | 12 | 6 |
| | 18 | 9 |
| | 24 | 12 |
| | 36 | 18 |
| | 48 | 24 |
| | | |
| 36/48/54 | 18 | 6 |
| | 24 | 9 |
| | 36 | 12 |
| | 48 | 18 |
| | 54 | 24 |

FIGURE 3.2: Different Acknowledgment rate corresponding to next suitable data rate

Careful observation of the Fig.3.2 reveals that the REACT algorithm can only notify the improved channel condition and switches to the next higher rate. On the other hand our acknowledgment rate mapping can signal multiple rate increase and decrease.

### 3.1.3 Receiver Side Mechanism Flow Diagram

The following Fig. 3.3 summarizes the steps in the receiver side.

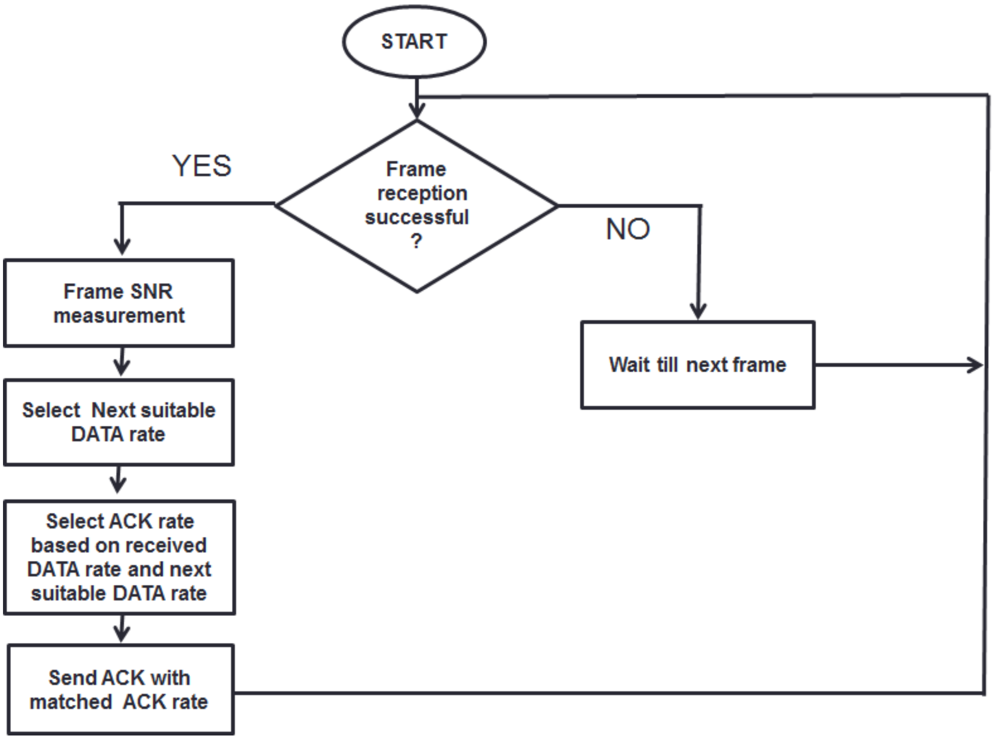



FIGURE 3.3: Receiver side mechanism flow diagram

## 3.2 NARC-Sender Side Mechanism

As mentioned in 1.4 our sender side mechanism is designed to encounter the stale feedback problem

### 3.2.1 Stepwise Procedure

Before every transmission say transmission **n** , the sender:

1 Overhears data while idle and calculates the SNR $(X_m)$.

2 Make an average of last 10 overheard data SNR and determines observed rate $(O_n)$ in that idle period.

3 Keeps track of the observed rates before its last 10 transmissions $(O_{n-1}, O_{n-2}, O_{n-3}, O_{n-4}, ......., O_{n-9}, O_{n-10})$.

4 Also keeps track of the receiver feedbacks after its last 10 transmissions $(F_{n-1}, F_{n-2}, F_{n-3}, F_{n-4}, ......., F_{n-9}, F_{n-10})$.

5 Determine a linear regression equation from its last 10 observed rates and receiver feedbacks.

6 Find the next transmission rate $R_n$ putting $O_n$ in the equation.

7 Transmit the data at $R_n$.

### 3.2.2 Demonstration

The sender side mechanism is demonstrated in the following Fig 3.4 :

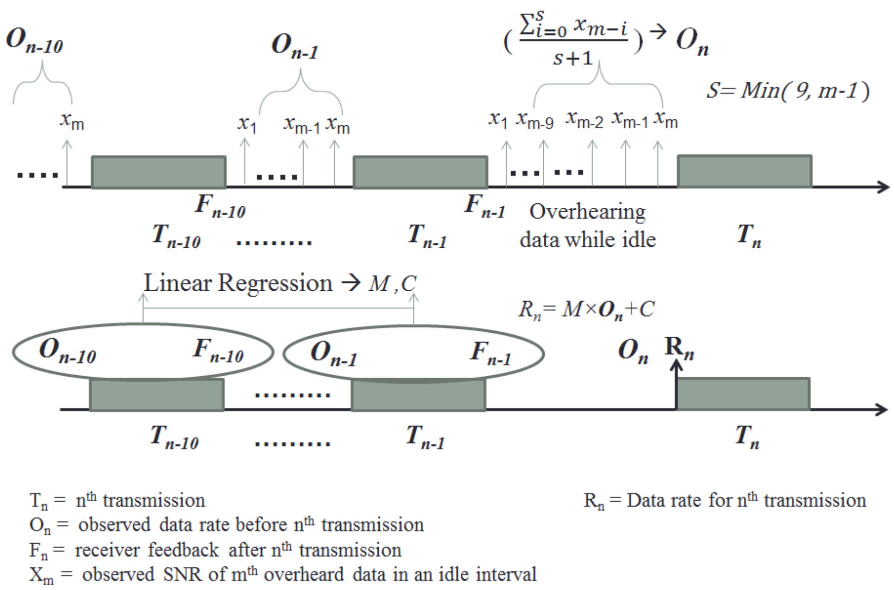



FIGURE 3.4: Demonstration of rate selection at sender side

As depicted in Fig. 3.4 the rectangular shaped structure represents the transmission from a particular node. The gaps between them represents the idle period before its transmission. Before the $n^{th}$ transmission $T_n$ it overhears the data from its neighbors and measures the SNR $X_m$ . Then from the average of last 10 overheard SNR values it finds the observed rate $O_n$ . Thus it will have a observed rate before its every transmission. How ever the node will keep track of its last 10 observed rates($O_{n-1}, O_{n-2}, O_{n-3}, O_{n-4}, ......., O_{n-9}, O_{n-10}$). Similarly it will also get a feedback from its receiver after every transmission. The node will also keep track of its last 10 feedback rates ($F_{n-1}, F_{n-2}, F_{n-3}, F_{n-4}, ......., F_{n-9}, F_{n-10}$). It is to mention that this feedback rates will be acknowledged by the receiver based on our receiver side algorithm as proposed in 3.1. At this stage the node will have pairs of observed rate and feedback rate from its last 10 transmissions ($T_{n-1}, T_{n-2}, T_{n-3}, T_{n-4}, ......., T_{n-9}, T_{n-10}$). Then the node will perform linear regression on these pairs of rates following the equations mentioned in 3.2.3. Finally it will put the latest observed rate $O_n$ in the equation and will determine the next transmission rate $R_n$ for $n^{th}$ transmission $T_n$ .

### 3.2.3 Linear regression equations

Linear regression [17] is a statistical technique to explore the relationship between two or more variables. For a set of paired data in our case-
$(O_{n-1}, F_{n-1}), (O_{n-2}, F_{n-2}), (O_{n-3}, F_{n-3}), ...., (O_{n-9}, F_{n-9}), (O_{n-10}, F_{n-10})$
It finds the best fitted lines equation for that set of data. The output is the value of the slope $M$ and intercept $C$ of that best fitted line.

For the last 10 observed SNR we use the limit $S$ as the number of overheard data can be less than 10.So it is obtained from the following equation-

$$S = min(9, m - 1)$$

and the observed rate is determined as-

$$\frac{\sum_{i=0}^{S}(x_{m-i})}{S+1} - - > O_n$$

Similarly for the limit of last 10 transmissions limit L is taken as -

$$limit, L = min(n - 1, 10)$$

And the slope $M$ is determined as -

$$M = \frac{\sum_{k=1}^{L}(F_{n-k}O_{n-k}) - \frac{\sum_{k=1}^{L}(F_{n-k})\sum_{k=1}^{L}(O_{n-k})}{L}}{\sum_{k=1}^{L}(O_{n-k})^2 - \frac{(\sum_{k=1}^{L}O_{n-k})^2}{L}}$$

Intercept $C$ is obtained from the equation -

$$C = \frac{\sum_{k=1}^{L}(F_{n-k})}{L} - (M * \frac{\sum_{k=1}^{L}(O_{n-k})}{L})$$

Finally putting $O_n$ in the line equation we get the next transmission rate $R_n$ by the following formula-

$$R_n = (M * O_n) + C$$

### 3.2.4 Initial Condition

- We use 24 Mbps as our initial transmission rate. Because according our acknowledgment method from Fig. 3.2 it allows to jump to higher rate or lower equally.
- In case of a station not getting any idle period before its first transmission $T_1$, we set the observation rate $O_1 = 24$ Mbps

### 3.2.5 Transmission failure

- Whenever a transmission gets failed we consider the corresponding feedback rate $F_n$ to be the next lower rate of transmitted rate $R_n$.
- And we send the next data with RTS protection.

# Chapter 4

# Simulation Results and Performance Evaluation

## 4.1 Simulation Setup

- We used ns-3 [18] as our simulator.
- We used custom topology for comparison purposes.
- Path loss model used: log-distance path-loss model [19].
- We have considered the transmission range up to 50 meters.
- All the simulations have been carried out for 100 seconds.
- We will perform simulation on the following scheme:
  -ARF
  -CARA
  -Ministrel
  -AMRR
  -RRAA
  -REACT
  -NARC

## 4.2 Performance Evaluation

We evaluated the performance of different existing algorithms with our proposed method NARC on the following criteria:

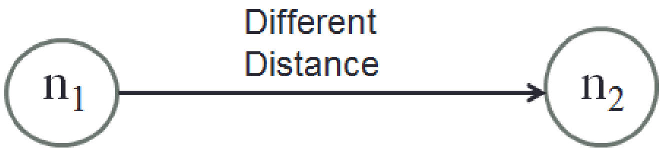



FIGURE 4.1: Toplogy of two static nodes

- Varying distance for static and mobile nodes
- Hidden node scenario
- Multiple flow scenario
- Varying the number of contending flows
- Varying the size of transmitted packets

### 4.2.1 Varying distance in static scenario:

#### 4.2.1.1 Topology

- **Number of nodes :** 2
- **Number of flows :** 1
- **Packet transmitted :** 100000
- **Packet size :** 2048 bytes
- **Flow data rate :** 54 Mbps
- **Path loss Model :** Log distance path loss model
- **Mobility Model :** Constant position mobility model
- **Initial Distance between nodes :** 5,10,15,20,25,30,35,40,45,50 meters

The above topology is depicted in Fig. 4.1

#### 4.2.1.2 Result

The Simulation on the above topology results in a graph illustrated in Fig. 4.2

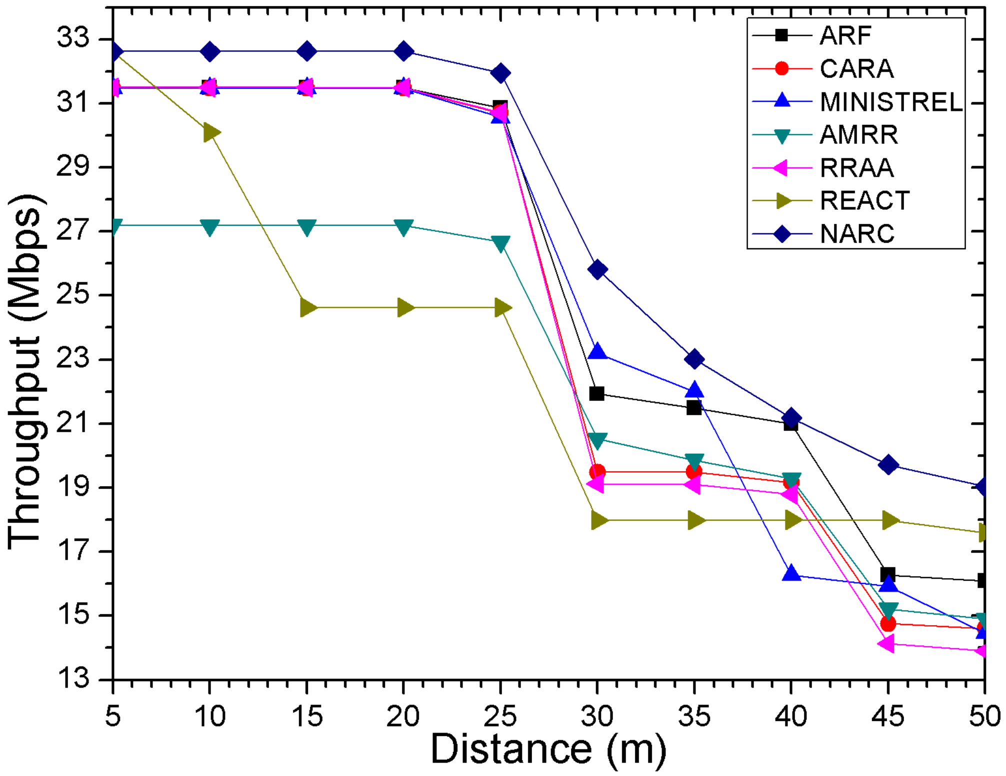



FIGURE 4.2: Throughput for various distance in static scenario

#### 4.2.1.3 Analysis

As we can see from the Fig. 4.2 clearly shows that the distance between the nodes has a great impact on the overall throughput of the network. Here we have keep the topology collision free and static therefore only the variation in the distance between nodes affects the throughput. As the figure suggests up to 25 meters all the algorithm shows comparatively better performance. However there is a drastic fall after 25 meter. The Frame based method ARF, CARA, Ministrel perform equally up to 25 meters but after that CARA and RRAA shows poor performance due to unnecessary use of RTS as there is no collision here. On the other hand ARF and Ministrel performs comparatively better than CARA and RRAA as they don't use RTS at all. Another frame based method AMRR performs poorly all throughout due to its conservative measure of rate selection.

SNR based method REACT though performs almost equal to NARC at 5 meter distance however it shows poor performance afterwards even poor than some other frame based method like ARF and Ministrel. The main reason is, it suffers from stale feedback problem heavily. And our proposed method NARC performs better than any other algorithm all throughout as it uses SNR based method and does not fall into the trap of stale feedback.

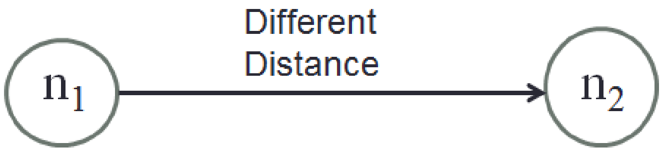



FIGURE 4.3: Toplogy of two mobile nodes

### 4.2.2 Varying distance in mobile scenario:

#### 4.2.2.1 Topology

- **Number of nodes :** 2
- **Number of flows :** 1
- **Packet transmitted :** 100000
- **Packet size :** 2048 bytes
- **Flow data rate :** 54 Mbps
- **Path loss Model :** Log distance path loss model
- **Mobility Model :** Randomwalk2d mobility model
- **Initial Distance**
  **between nodes :** 5,10,15,20,25,30,35,40,45,50 meters

The above topology is depicted in Fig. 4.3

#### 4.2.2.2 Result

The Simulation on the above topology results in a graph illustrated in Fig. 4.4

#### 4.2.2.3 Analysis

Here in this Fig. 4.4 we have considered the effects of mobility in the algorithms' performance. As the figure depicts the algorithms follow almost a similar trend like the one in Fig. 4.2. The rationale is same as mentioned in 4.2.1.3 .

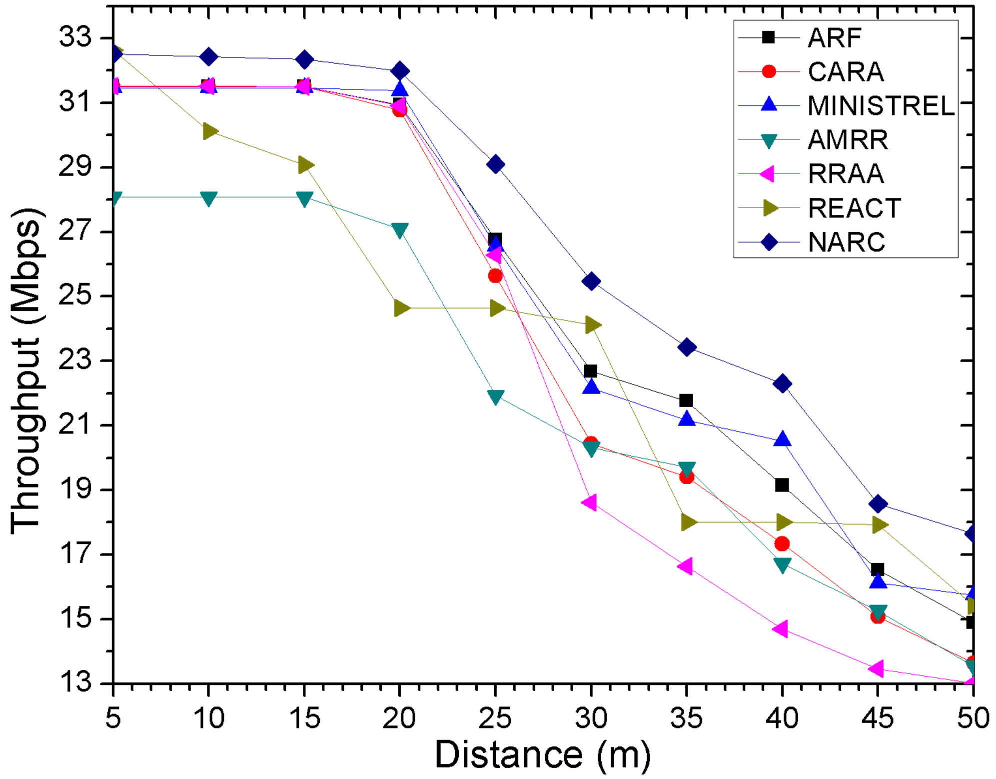



FIGURE 4.4: Throughput for various distance in mobile scenario

### 4.2.3 Hidden node scenario:

#### 4.2.3.1 Topology

- **Number of nodes :** 3
- **Number of flows :** 2
- **Packet transmitted :** 100000
- **Packet size :** 2048 bytes
- **Flow data rate :** 54 Mbps
- **Path loss Model :** Log distance path loss model
- **Mobility Model :** Constant position mobility model

The above topology is depicted in Fig. 4.5

#### 4.2.3.2 Result

The Simulation on the above topology results in a graph illustrated in Fig. 4.6

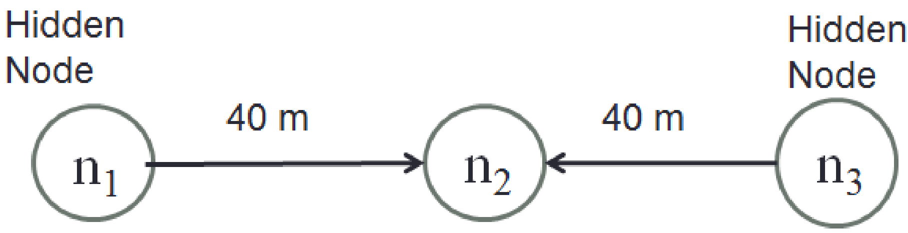



FIGURE 4.5: Hidden node topology

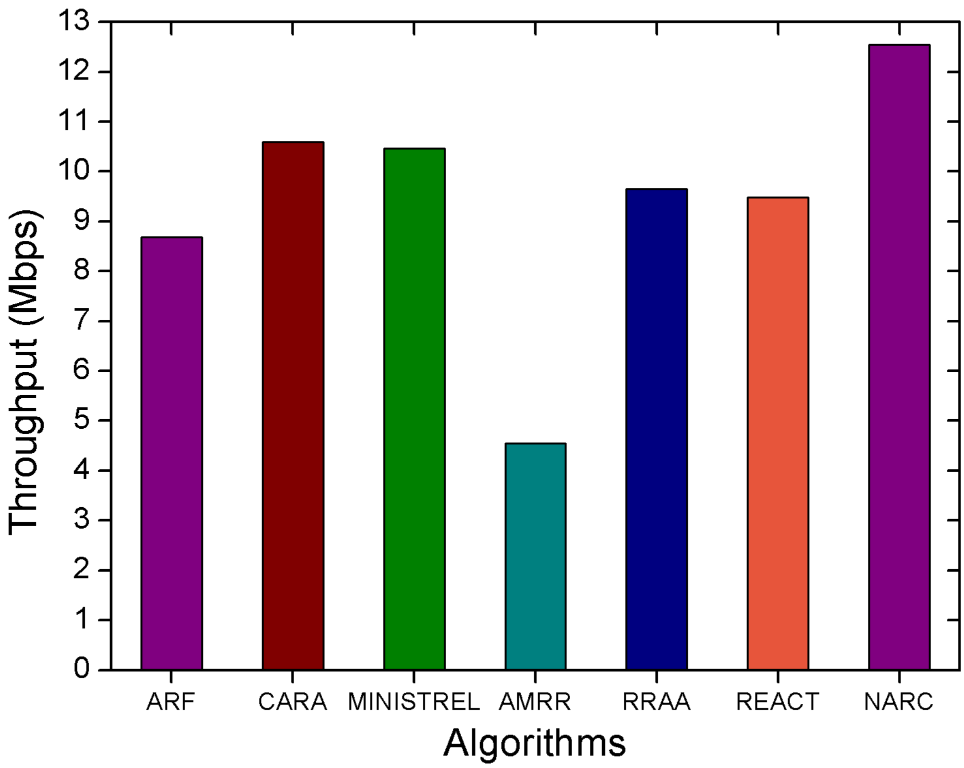



FIGURE 4.6: Throughput for various distance in mobile scenario

#### 4.2.3.3 Analysis

In the Fig. 4.6 we have considered a hidden node scenario. As the figure illustrates the algorithms with RTS protection performs better in this scenario. The frame based method with RTS protection like CARA, RRAA performs better than ARF and AMRR. However Ministrel though don't use the RTS frame to differentiate the cause of frame loss still it performs quite better due to its fast rate switching mechanism.

On the other hand REACT due to its sequential approach and being unable to tackle the stale feedback problem shows less performance. However our method NARC outperforms every other algorithm in this case also.

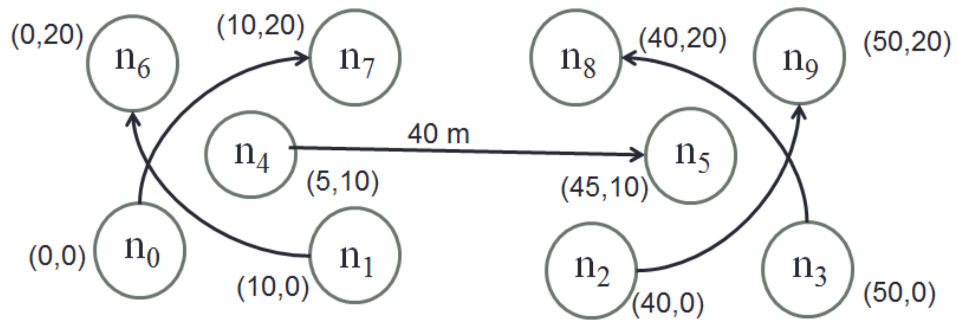



FIGURE 4.7: Multiple flows topology

### 4.2.4 Multiple flows scenario:

#### 4.2.4.1 Topology

- **Number of nodes :** 10
- **Number of flows :** 5
- **Packet transmitted :** 100000
- **Packet size :** 2048 bytes
- **Flow data rate :** 54 Mbps
- **Path loss Model :** Log distance path loss model
- **Mobility Model :** Constant position mobility model

The above topology is depicted in Fig. 4.7

#### 4.2.4.2 Result

The Simulation on the above topology results in a graph illustrated in Fig. 4.8

#### 4.2.4.3 Analysis

To consider the performance of different algorithms in a congested network we have created 5 flows among 10 stations in a close proximity as depicted in Fig. 4.7. And the corresponding result presented in Fig. 4.8 shows that our algorithm gives much better performance even in the case of congested network. The reason is, in congested network the channel condition varied rapidly and to cope up with dynamic channel is the main challenge. As we use SNR based method where the receiver can inform the sender

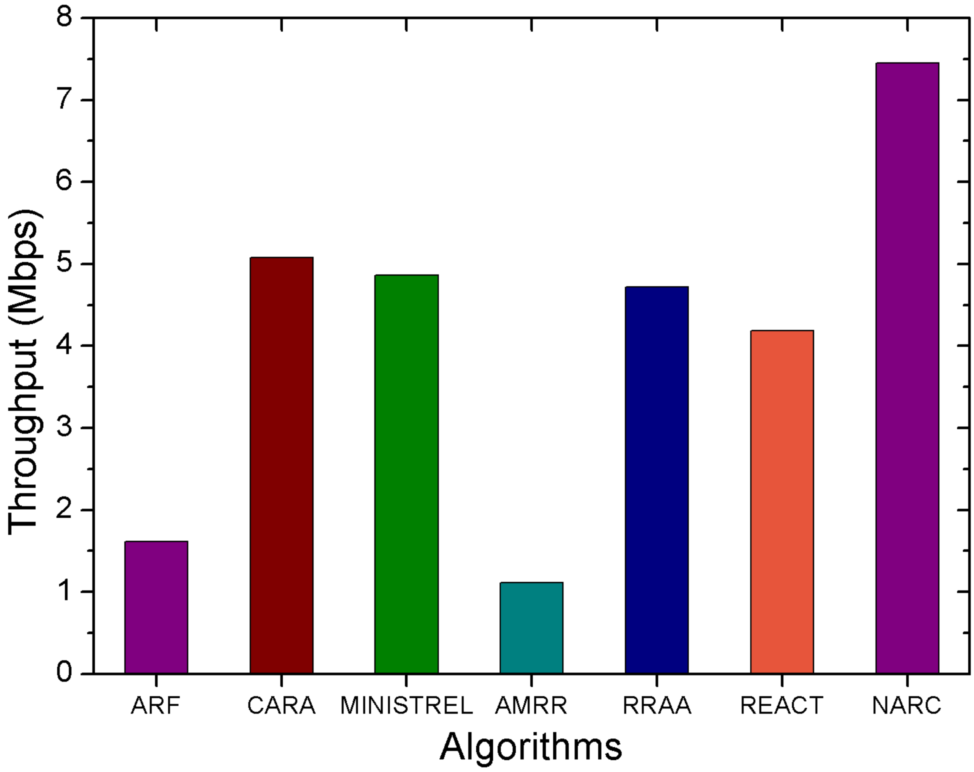



FIGURE 4.8: Throughput for multiple flow scenario

about the most accurate channel condition coupled with the sender side mechanism of overhearing data.That's why NARC gives the best performance compared to the other algorithms.

### 4.2.5 Multiple flows at sender side:

#### 4.2.5.1 Topology

- **Number of nodes :** 6
- **Number of flows :** 3
- **Packet transmitted :** 100000
- **Packet size :** 2048 bytes
- **Flow data rate :** 54 Mbps
- **Path loss Model :** Log distance path loss model
- **Mobility Model :** Constant position mobility model

The above topology is depicted in Fig. 4.9

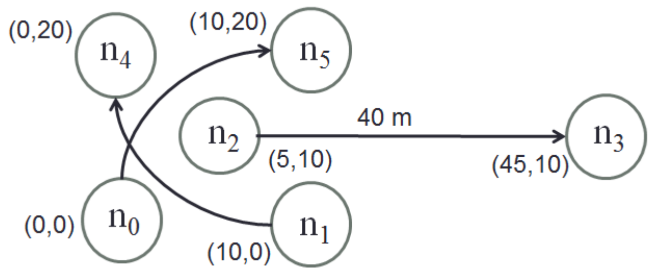



FIGURE 4.9: Topology of multiple flows at sender side

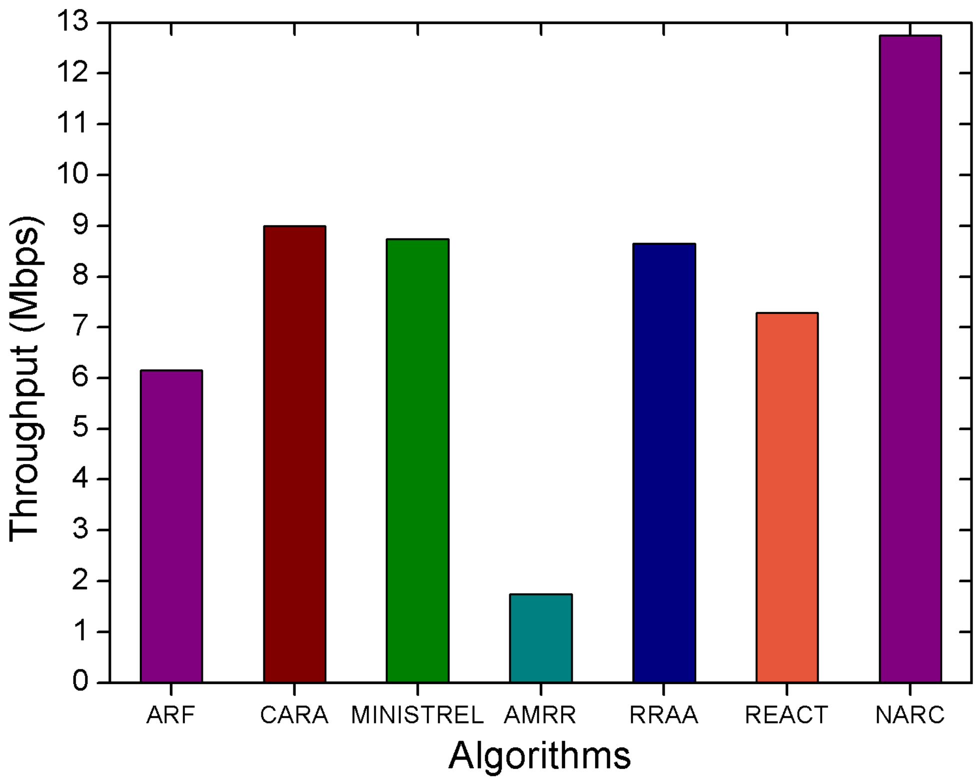



FIGURE 4.10: Throughput for multiple flows at sender side

#### 4.2.5.2 Result

The Simulation on the above topology results in the graphs illustrated in Fig. 4.10 and Fig. 4.11

#### 4.2.5.3 Analysis

To further verify the performance of different algorithms when there is uneven congestion at the sender and receiver side we created a topology depicted in Fig. 4.9 . We designed the topology in such a that there is more congestion at the sender side than the receiver.

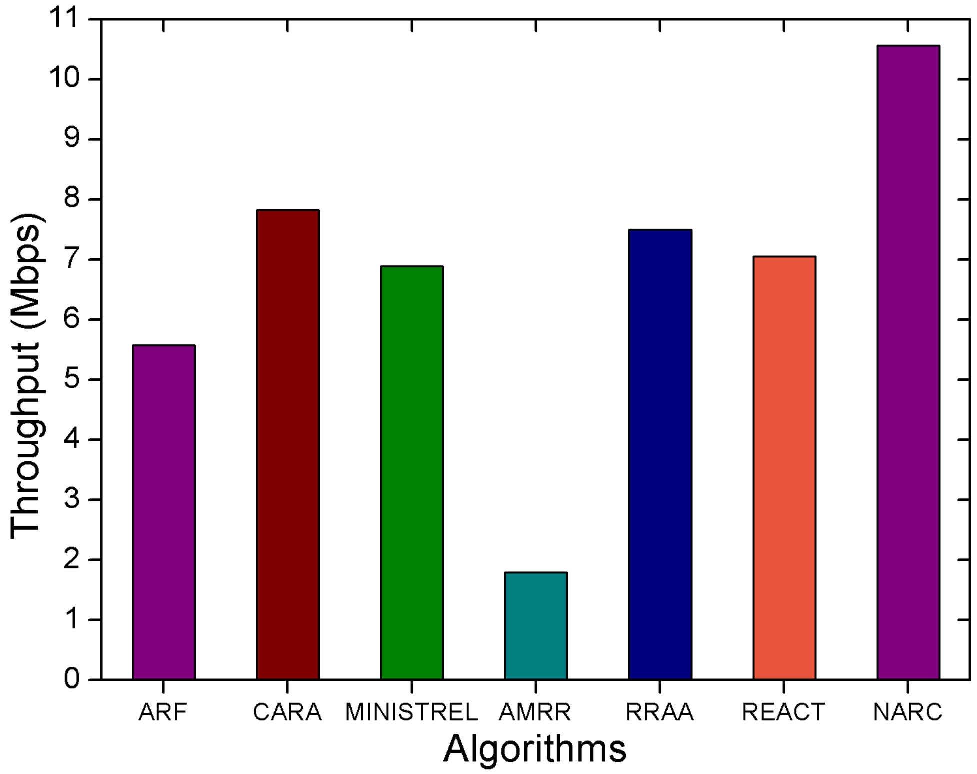



FIGURE 4.11: Flow throughput from $n_2$ to $n_3$

As our algorithm has two parts one is sender side and another is receiver side, we wanted to verify our algorithm's performance in uneven scenario. As shown in Fig. 4.10 NARC gives better performance than others by remarkable margin. Moreover we also observed the throughput of the weakest flow from $n_2$ to $n_3$ in Fig. 4.11. In this case also, the weakest flow has the maximum throughput compared to others.

### 4.2.6 Multiple flows at receiver side:

#### 4.2.6.1 Topology

- **Number of nodes :** 6
- **Number of flows :** 3
- **Packet transmitted :** 100000
- **Packet size :** 2048 bytes
- **Flow data rate :** 54 Mbps
- **Path loss Model :** Log distance path loss model
- **Mobility Model :** Constant position mobility model

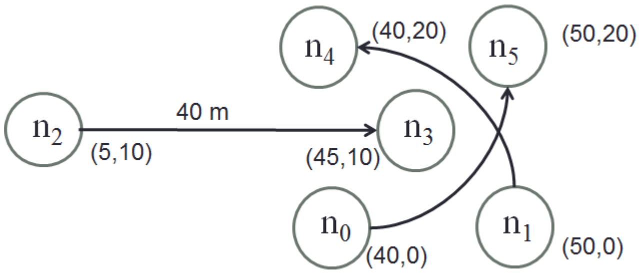



FIGURE 4.12: Topology of multiple flows at receiver side

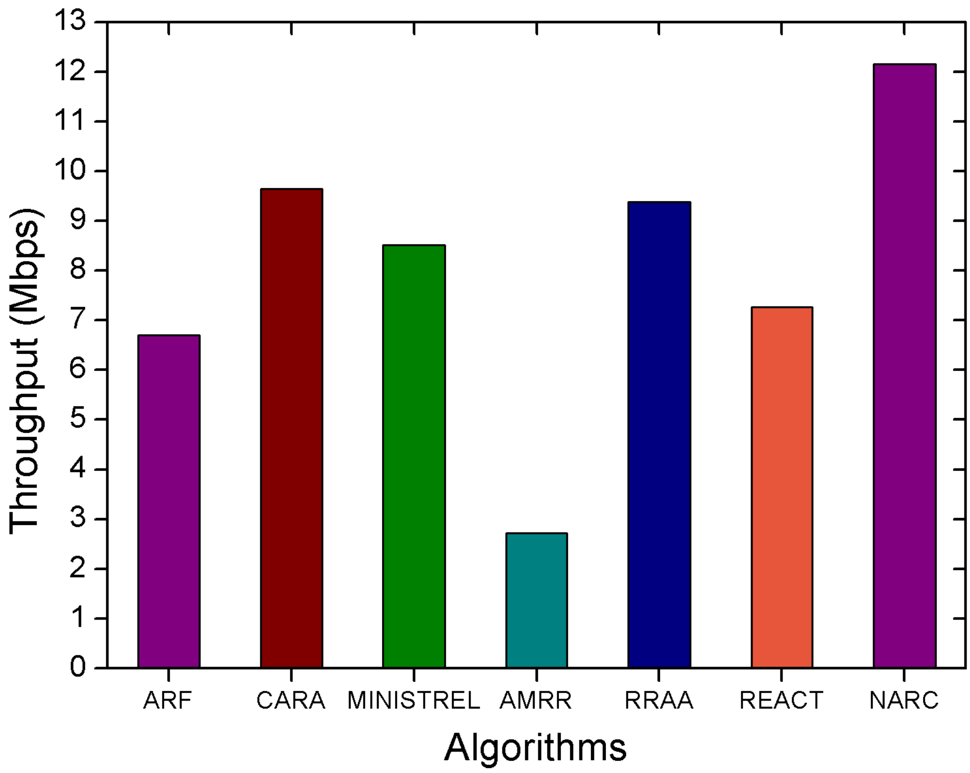



FIGURE 4.13: Throughput for multiple flows at receiver side

The above topology is depicted in Fig. 4.12

#### 4.2.6.2 Result

The Simulation on the above topology results in the graphs illustrated in Fig. 4.13 and Fig. 4.14

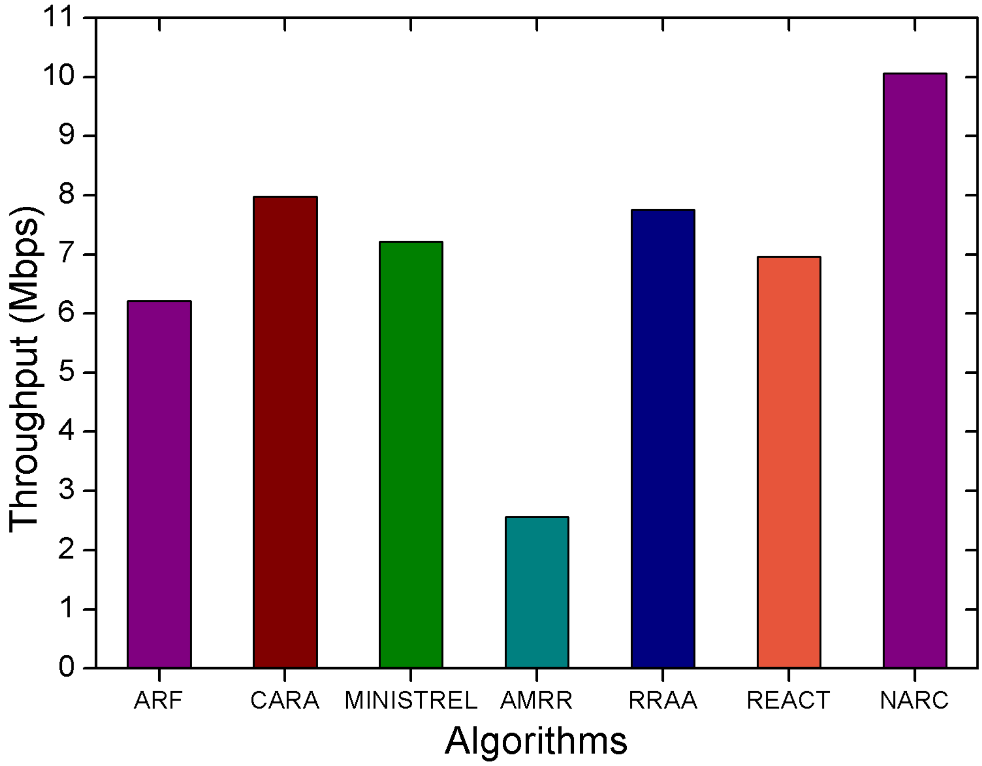



FIGURE 4.14: Flow throughput from $n_2$ to $n_3$

#### 4.2.6.3 Analysis

This time we created more congestion at the receiver side as shown in Fig. 4.12. Like the one in 4.10 NARC performs better in this case also. The rationale is, owing to accurate feedback from receiver our algorithm can predict the next suitable rate more precisely thereby results in increased throughput as presented in Fig. 4.13. Also for the weakest flow from $n_2$ to $n_3$ our algorithm shows improved performance as illustrated in Fig. 4.14.

### 4.2.7 Varying number of contending flows:

#### 4.2.7.1 Topology

- **Number of nodes :** 9
- **Number of flows :** 1 to 8
- **Packet transmitted :** 100000

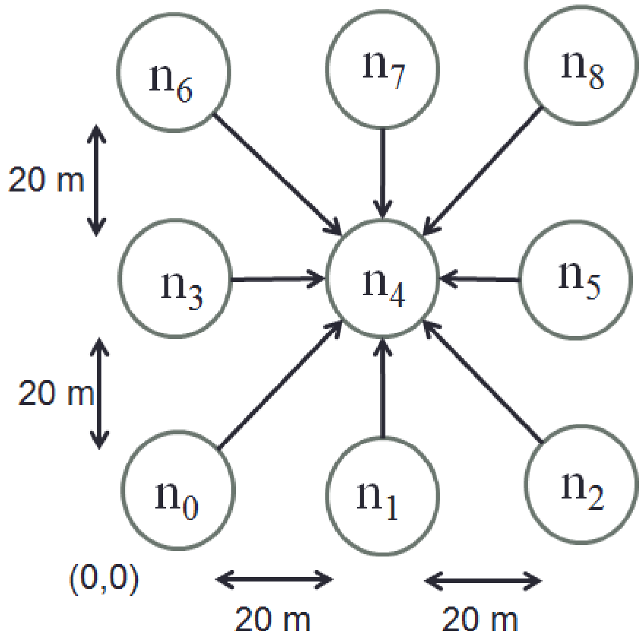



FIGURE 4.15: Topology for various number of contending flows

- **Packet size :** 2048 bytes
- **Flow data rate :** 54 Mbps
- **Path loss Model :** Log distance path loss model
- **Mobility Model :** Constant position mobility model

The above topology is depicted in Fig. 4.15

#### 4.2.7.2 Result

The Simulation on the above topology results in the graph illustrated in Fig. 4.16

#### 4.2.7.3 Analysis

Here we tried to evaluate the performance of various algorithms with different number of contending flows. To serve this purpose we placed 9 nodes in a grid topology and made the central node as the common sink. Other 8 nodes act as the source as shown in Fig. 4.15. At first we simulated for 1 flow then 2 flows likewise up to 8 flows. This is a scenario of highly congested topology and in this case also our algorithm shows a improved performance by a certain degree as depicted in Fig. 4.16. The performance of each algorithm decreases gradually with the increasing number of contending flows. Still

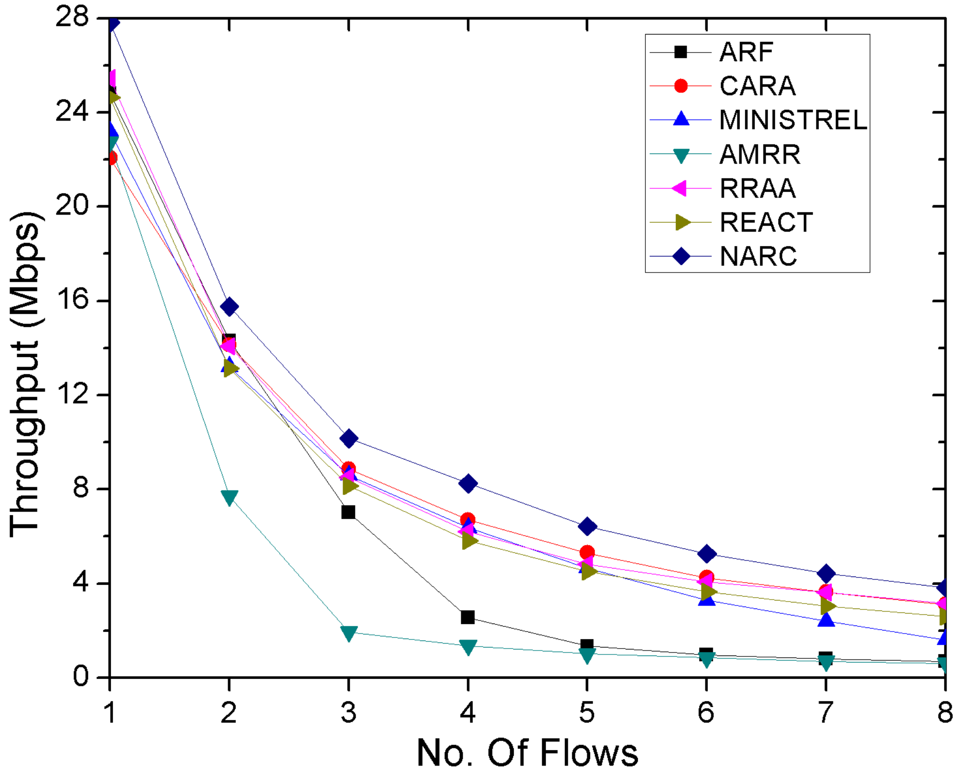



FIGURE 4.16: Throughput for various number of contending flows

NARC manages to outperform every other algorithm for different number of contending flows.

### 4.2.8 Varying packet size:

#### 4.2.8.1 Topology

- **Number of nodes :** 2
- **Number of flows :** 1
- **Packet transmitted :** 100000
- **Packet size :** 250, 500, 1000, 1500, 2000, 2048 bytes
- **Flow data rate :** 54 Mbps
- **Path loss Model :** Log distance path loss model
- **Mobility Model :** Constant position mobility model
- **Initial Distance between nodes :** 25 meters

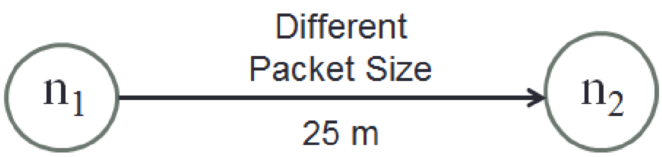



FIGURE 4.17: Topology of two nodes with different packet size

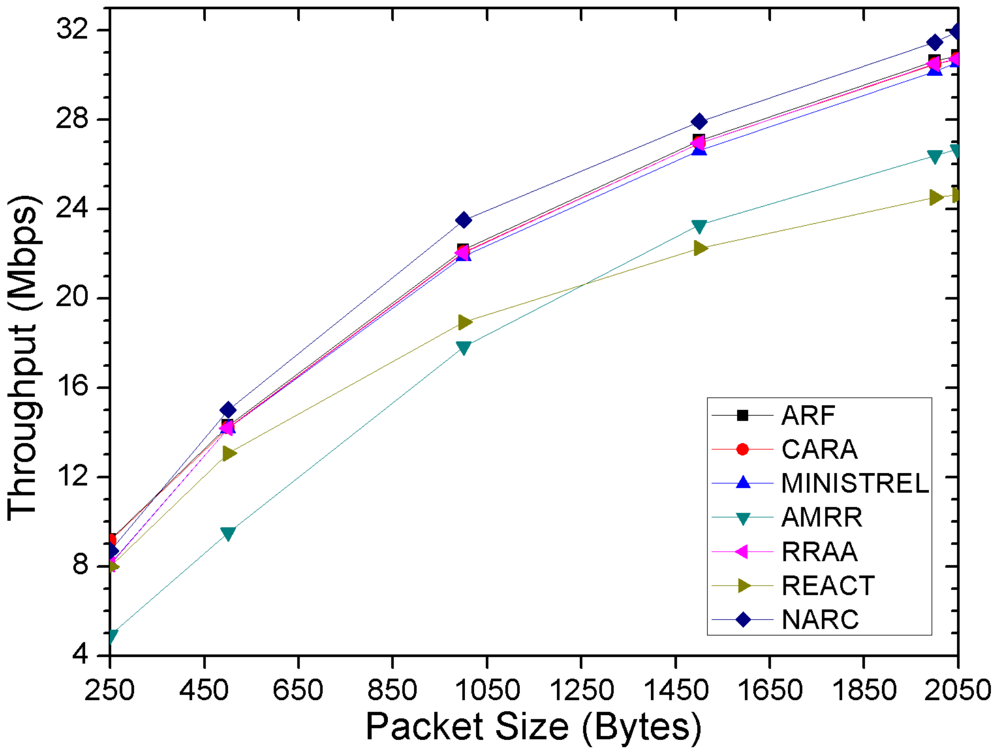



FIGURE 4.18: Throughput for various packet size in static scenario

The above topology is depicted in Fig. 4.17

#### 4.2.8.2 Result

The Simulation on the above topology results in a graph illustrated in Fig. 4.18

#### 4.2.8.3 Analysis

Finally we evaluated the performance of our algorithm for different size of packets. Careful insight to the Fig. 4.18 reveals that with the increasing packet size the throughput also increases. And this case also NARC leads every other algorithm in terms of throughput gain.

# 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 SNR based rate adaptation scheme NARC in which receiver controls the ACK(Acknowledgment) transmission rate as a means to dictate the sender to adjust data rate. NARC is also responsive to time-varying wireless channel owing to the accurate and instant feedback. Further, it mitigates stale feedback problem through overhearing the neighborhood data and attempts to transmit packet at the rate most suitable to the channel quality. Finally it can switch to optimal transmission rate. Moreover, it uses RTS adaptively in order to avoid the rate avalanche effect and also ensuring minimum overhead.

The characteristics of our algorithm along with other existing ones are summarized in the following Fig. 5.1. Green and red colors represent the pros and cons of different algorithms.

## 5.2 Future Work

We can see clearly from our simulation results that NARC outperforms all existing algorithms especially in dynamic channel conditions. To further verify the robustness of our algorithm we will test it in real life scenario using Mad-WiFi device driver [20].

| Algorithm | Type | Rate Switch | Channel Quality estimation at | RTS/CTS Use | RTS/CTS Overhead | Stale Feedback |
|---|---|---|---|---|---|---|
| ARF | Frame based | Sequential | Sender | No | No | N/A |
| CARA | Frame based | Sequential | Sender | Adaptive | Minimum | N/A |
| Samplerate | Frame based | Random | Sender | No | No | N/A |
| Ministrel | Frame based | Random | Sender | No | No | N/A |
| AMRR | Frame based | Sequential | Sender | No | No | N/A |
| RRAA | Frame based | Sequential | Sender | Adaptive | Minimum | N/A |
| RBAR | SNR based | Optimal | Receiver | Always | Maximum | No |
| REACT | SNR based | Sequential | Receiver | Adaptive | Minimum | Yes |
| NARC | SNR based | Optimal | Both | Adaptive | Minimum | No |

FIGURE 5.1: Rate adaptation algorithms in a nutshell

# Bibliography


[1] IEEE Standard for Information technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements -Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, *IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999)*, June 12, 2007.

[2] http://www.webopedia.com/TERM/8/802.11.html; Accessed 12.11.2015.

[3] https://en.wikipedia.org/wiki/Signal-to-noise ratio; Accessed 12.11.2015.

[4] Behrouz A Forouzan, *Data Communications and Networking* $4^{th}$ *edition*, McGraw-Hill Education, Chapter 12 .

[5] Behrouz A Forouzan, *Data Communications and Networking* $4^{th}$ *edition*, McGraw-Hill Education, Chapter 14 .

[6] Liqiang Zhanga,Yu-Jen Chenga, Xiaobo Zhoub, "Rate avalanche: Effects on the performance of multi-rate 802.11 wireless networks ", *Performance Modeling and Evaluation of Telecommunication Systems, Elsevier*, Volume 17, Issue 3, March 2009, Pages 487–503.

[7] A. Kamerman and L. Monteban, "WaveLAN-II: a high-performance wireless LAN for the unlicensed band ", *Bell Labs Technical Journal*, vol.2, no.3, pp. 118–133, 1997.

[8] J. Kim, S. Kim, S. Choi, and D. Qiao. "CARA: Collision-aware Rate Adaptation for IEEE 802.11 WLANs ", *In Proc. INFOCOM*, Barcelona, April 23–29, 2006.

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

[10] Minstrel Linux Wireless. http://linuxwireless.org/en/developers /Documentation/mac80211/RateControl/minstrel; Accessed 25.10.2015.

[11] M. Lacage, M. Manshaei and T. Turletti, "IEEE 802.11 Rate Adaptation : A Practical Approach ", in *MSWiM04*, 2004, pp. 126–134.

[12] S. Wong, H. Yang, S. Lu, and V. Bharghavan, “Robust Rate Adaptation for 802.11 Wireless Networks”, in *Proc. MOBICOM*, Los Angeles, CA, USA, 2006.

[13] Chang-Woo Ahn, Sang-Hwa Chung, “Enhancing WLAN Performance with Rate Adaptation Considering Hidden Node Effect ”,*International Journal of Distributed Sensor Networks*, Volume 2015.

[14] G.Holland , N.Vaidya, and P.Bahl ,“A rate-adaptive MAC protocol for multi-hop wireless networks ”, in *ACM MOBICOM'01*, Rome, Italy, 16-21 July, 2001.

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

[16] D. Qiao, S. Choi, 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.

[17] Douglas C. Montogomery , George C. Runger, “Applied Statistics and Probability for Engineers ”, Third Edition, John Wiley & Sons, Chapter 11.

[18] “ns-3:The-Network-Simulator-3,”2008.[Online]-Available: https://www.nsnam.org/ ; Accessed 12.10.2015.

[19] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed., Upper Saddle River, NJ: Prentice Hall, PTR, 2002.

[20] “Madwifi-Project-Information.”2004.[Online]-Available: https://sourceforge.net/projects/madwifi ; Accessed 19.11.2015.