UNIVERSITY OF CINCINNATI

Date: 23-Feb-2010

I, Ramakrishnan Kripakaran, hereby submit this original work as part of the requirements for the degree of:

Master of Science

in Computer Science

It is entitled:

Effective Strategies for Mesh Router Selection in Wireless Mesh Networks

Student Signature: Ramakrishnan Kripakaran

This work and its defense approved by:

Committee Chair: Dharma Agrawal, DSc
Dharma Agrawal, DSc

Fred Annexstein, PhD
Fred Annexstein, PhD

Chia Han, PhD
Chia Han, PhD

2/26/2010
Effective Strategies for Mesh Router Selection in Wireless Mesh Networks

A thesis submitted to
Division of Research and Advanced Studies
of the University of Cincinnati
in partial fulfillment of the
requirements for the degree of

Master of Science

in the Department of Computer Science
of the College of Engineering
February, 2010
by

Ramakrishnan Kripakaran

Bachelor of Engineering (Computer Science & Engineering)
Bharathiar University, India.
June 2003

Thesis Advisor and Committee Chair: Dr. Dharma P. Agrawal
Abstract

Wireless Mesh Network (WMN) is a developing technology which envisions a wireless backbone architecture (in place of the existing wired) to aid in providing internet connectivity to users at the residential and commercial level. In a WMN, the density of Mesh Clients (MCs) and the rate of traffic generated from each client are not uniformly distributed. Also, the traffic routes from a Mesh Router (MR) to the Internet GateWay (IGW) are multi-hop and multi-path in nature. Due to these factors, each MR has a different performance. For large scale WMN, maintaining such information about each MR in a centralized manner is not a practical and scalable solution. In addition, when new MCs enter the WMN they have little or no information about the performance of MRs. With scarce information, it is difficult for a MC to select a MR which can provide the best service. Therefore, MCs must directly request each MR in its vicinity to obtain information that can help make this decision. Since MCs have the freedom to select a MR of their choice, examining all nearby MR increases the chance of selecting a better MR suited for their application. However, often a MC has limited time available for selecting the MR. We propose three strategies that a MC can implement in making the selection of the MR – opportunistic, conservative and calculatingly-opportunistic. We study the performance of these strategies theoretically and statistically. We obtain the closed form solution for the conservative strategy which proves to be the best of the three strategies. We also identify the network parameter that can be used to rank the performance of an MR. We perform extensive network simulations to evaluate the performance of the strategies with various network configurations.
Acknowledgements

I would like to thank my advisor Dr. Dharma P. Agrawal for providing me the opportunity to do research under his guidance and for his support and motivation throughout the course of my Masters program. I would also like to thank Dr. Fred Annexstein and Dr. Chia Han for reviewing my thesis and for being part of my thesis defense committee.

I would like to thank fellow researchers at the Center for Distributed and Mobile Computing (CDMC) for their support and suggestions. I especially thank Peter Jun for his ideas and help at various phases of my research.

I thank all my friends, in Cincinnati and outside, for their help and motivation. Lastly, I thank my family for their constant support and encouragement during the entire period of my stay in Cincinnati.
# Table of Contents

## CHAPTER 1

1. Introduction  
   1.1 Wireless Mesh Networks  
   1.2 Characteristics of a WMN  
   1.3 Problem Definition  
   1.4 Thesis Organization  

## CHAPTER 2

2. Related Research  
   2.1 Selection Problem strategies and applications  

## CHAPTER 3

3. Proposed Strategies  
   3.1 Naive Strategy  
   3.2 Opportunistic strategy  
   3.3 Conservative strategy  
   3.4 Calculatingly-Opportunistic strategy  

## CHAPTER 4

4. Simulation Results  
   4.1 Numerical Analysis  


CHAPTER 5

5. Conclusion and Future work

5.1 Conclusion

5.2 Future Work

BIBLIOGRAPHY
List of Figures

Figure 1.1 Architecture of a Wireless Mesh Network ............................................................................ 2
Figure 3.1 Flowchart for the Opportunistic strategy ........................................................................... 15
Figure 3.2 Example of the operation of the Opportunistic Strategy ............................................................. 16
Figure 3.3 Flowchart for the Conservative strategy ............................................................................ 24
Figure 3.4 Example of the operation of the Conservative Strategy .............................................................. 25
Figure 3.5 Flowchart for the Calculatingly-Opportunistic strategy.............................................................. 27
Figure 3.6 Example of the operation of the Calculatingly-Opportunistic Strategy ........................................ 28
Figure 4.1 Probability of selecting the best, n = 100 .......................................................................... 30
Figure 4.2 Probability of selecting one of the top three maximum ................................................................ 31
Figure 4.3 Experimental Setup .............................................................................................................. 35
Figure 4.4 Execution Flow .................................................................................................................... 36
Figure 4.5 Snapshot of a WMN in Grid topology .................................................................................. 38
Figure 4.6 Snapshot of a WMN in Random topology ............................................................................ 39
Figure 4.7 Results for AODV, traffic rate = 256 bytes/s ............................................................................ 41
Figure 4.8 Results for AODV, traffic rate = 341 bytes/s .......................................................................... 42
Figure 4.9 Results for AODV, traffic rate = 512 bytes/s .......................................................................... 43
Figure 4.10 Results for Bellman-Ford, traffic rate = 256 bytes/s ............................................................. 44
Figure 4.11 Results for Bellman-Ford, traffic rate = 341 bytes/s ............................................................. 45
Figure 4.12 Results for Bellman-Ford, traffic rate = 512 bytes/s ............................................................. 46
Figure 4.13 Results for AODV, traffic rate = 256 bytes/s ...................................................................... 47
Figure 4.14 Results for AODV, traffic rate = 512 bytes/s ...................................................................... 48
Figure 4.15 Results for Bellman-Ford, traffic rate = 256 bytes/s ............................................................. 49
Figure 4.16 Results for Bellman-Ford, traffic rate = 512 bytes/s ............................................................. 50
Chapter 1

1. Introduction

1.1 Wireless Mesh Networks

A Wireless Mesh Network (WMN) is a multi-channel, multi-hop wireless network, comprising of different types of nodes, namely, Internet GateWay (IGW), Mesh Routers (MRs) and Mesh Clients (MCs). The architecture of a WMN is very similar to the Wireless Local Area Network (WLAN) with a major exception that the backbone network except the IGWs, is also wireless in a WMN [1]. WMN is seen as the better alternative to the traditional WLAN network because of its easy and cheap deployment. IGWs have both wireless and wired interfaces. Based on the network architecture, WMNs can be classified into three types - Infrastructure/Backbone WMN, Client WMN and Hybrid WMN [2]. Infrastructure/Backbone WMN is the most common used type. Figure 1.1 shows the architecture of an infrastructure WMN.

The three types of network devices are shown in Figure 1.1,

- The IGW has a wired connection to the Internet backbone. There could be more than one IGW when the WMN is large. Multiple IGWs help in balancing the load in the network [3] and in achieving higher network throughput [4]. The IGW can be the same router as an MR or a more powerful one. They are also called Mesh Points (MPs) [1].
The MRs form the backbone of the WMN. They are interconnected wirelessly. Every MR can either act as an active or a passive router, i.e., it can act as a wireless access point to serve clients or simply be a forwarding agent for other MRs connected to it either directly or in a multi-hop fashion.

The MCs are the devices which make use of the WMN for getting connected to the internet. This could be any device that is equipped with wireless Network Interface Cards (NICs) like laptops, phones, PDAs.

While WMNs are derived from the concepts of mobile adhoc network (MANET) [5], there are certain significant differences between WMN and MANETs.
1.2 Characteristics of a WMN

The major characteristics of a WMN are described below.

- **Multi-hop network:** WMN is a multi-hop network. This means a MC connects to the internet or the IGW through a series of MRs in a multi-hop fashion. The MRs communicate wirelessly which makes it easier to deploy the network. The advantage is that there is no need for line of sight to the IGW by the client as long as it can connect to one of the MRs. Due to the multi-hop nature of the network, it is important to choose an efficient routing protocols for the wireless backbone [6].

- **Multi-channel network:** Every MR in the WMN network is equipped with more than one network interface card (NIC), thereby enabling the MR to operate on orthogonal multiple channels simultaneously. Therefore, an MR can communicate with multiple MRs at the same time which might increase the throughput of the network [7, 8].

- **Self-configuration:** The MRs in a WMN are capable of learning their neighbors and the links to the other nodes. Therefore, there is no need for configuration at each node. This reduces the deployment time. Also, scalability is an important factor for multi-hop networks [9]. The self-configuration property of WMN makes the network easily scalable.

- **Multi-path network:** The path from any MR to the IGW is not a fixed one. That is, all packets sent from an MR to the IGW (or from IGW to an MR) can take different paths to reach the IGW. This means that if there is a link breakdown in the backbone network, the packets can still reach the IGW by taking an alternate route. This is called the *self-healing* characteristic of WMN since the MRs can learn dynamically the health of their
neighbors and associated links. Through effective use of multi-path routing, it is possible to achieve load balance and thereby increasing the network throughput [10].

- **No power constraints:** There are no power restrictions for a MR in a WMN as in a sensor network, where power consumption plays a major role. This is because usually the MRs are immobile and therefore always placed where connection to a power source is feasible.

- **Traffic pattern:** For the most part, the traffic in a WMN is between the MRs and the IGW. The paths leading to the IGW are more constrained and thereby, the load is more on the MRs that are nearer to the IGW.

### 1.3 Problem Definition

In a WLAN, the backbone network which connects the Access Points (APs) to the Internet is a wired network and the access medium connecting the clients and the APs is wireless. Because of numerous issues in the wireless connectivity (like interference, channel contention, bandwidth-availability), the service received by the clients is more dependent on the wireless connectivity than the backbone network. In the case of a WMN, since the backbone network is connected wirelessly, the service obtained by the clients of the WMN is affected by both the MC-MR connectivity at the end-user level as well as the MR-MR connectivity in the backbone. Also, in a WMN, the density of MCs and the traffic generated by each MC is not uniformly distributed. This leads to unbalanced network with each MR having a different performance. It is also not feasible to keep track of this information in a centralized manner because of the dynamic nature of the WMN and the traffic overhead this might add to the existing network.
Therefore, when new MCs enter the WMN, they have limited or no information about the network. The time available for the MC to make the decision is limited. Therefore, given a time interval, the MCs can poll the MRs in the vicinity for more information and then make the selection.

The problem can be defined mathematically as follows. Let $f(r)$ be the function that represents the ranking policy followed by a MC. It takes information provided by an MR, $r \in R$, where $R$ is the set of all MRs and returns the rank for MR $r$. For instance, $f(r)$ returns a lower value to indicate that MR $r$ will not be able to provide good quality of service. Then, the objective is to find the MR $r^*$ such that $f(r^*)$ is maximized, i.e., $f(r^*) = \max_{r \in R} f(r)$. This follows the constraint that there is an upper bound to the time ($t$) that can be spent in probing the MRs and the probing happens linearly in the forward or reverse direction. In addition, it is assumed that the ranking values remain constant over $t$. In order to simplify the problem details, the time constraint is mapped to the number of MRs that a MC can visit before making the selection, i.e., $t = k \Delta t$, where $k$ is the number of MRs that can be visited by a MC before making the selection and $\Delta t$ is the constant time that a MC takes to move to the next or the previous MR.

The following are the contributions in this thesis

- We propose three simple distributed probabilistic strategies to solve the above defined problem under a time-constraint and known ranking values of the MRs. The strategies do not require prior knowledge about the distribution of the ranking values.
- We analyze the performance of each strategy through theoretical and statistical means. We provide a closed form solution for the conservative strategy which proves to be the best of the three strategies.
- We analyze various network parameters and identify the one that can be used for ranking the MRs.
- We apply proposed strategies on different configurations of WMN using a network simulator and evaluate their performance.

1.4 Thesis Organization

The thesis is divided into several chapters described as follows. Chapter 2 details the related work that is available in the area of selection problems and their applications in the wireless network field. Chapter 3 describes the approach and the algorithms for the three strategies proposed to solve the problem, namely the opportunistic, conservative and calculatingly-opportunistic strategy. The theoretical analysis of the conservative strategy is also described in detail. Chapter 4 details on the different simulations done to evaluate the strategies. This includes the statistical and network analysis. The statistical analysis runs several numerical simulations over random numbers to determine which strategy is the most efficient among the three. This is used to verify the validity of the theoretical result that was arrived for the conservative strategy. This chapter also provides a description on the selection of the network parameter that is used for ranking in the algorithm. As part of the network analysis, the chapter describes in detail the experimental setup (tools) and the simulation setup (network configurations) used in the simulation followed by the results obtained and a summary on the
results. Chapter 5 concludes the thesis and considers possible extensions that are possible on the current work.
Chapter 2

2. Related Research

The main aim of this thesis is to propose a distributed strategy for the selection problem that can work without any prior knowledge. Selection problem is a very common problem that applies to a multitude of areas. Moreover, there could be a lot of variations in the problem (in the form of constraints) that make each problem unique. This chapter will introduce some of the existing work in this area.

2.1 Selection Problem strategies and applications

Kanodia [11] present a protocol for choosing a best channel from a spectrum of available channels. The protocol is named Multi-channel Opportunistic Auto-Rate media access protocol (MOAR). The strategy proposed in this protocol is to probe each channel in the spectrum and assign a constant value which represents the quality of the channel being probed. They select the best channel by using an optimal skipping rule, which is to select the first channel that is better than a pre-defined threshold value of Signal to Noise Ratio (SNR). The variation from our current selection problem (apart from the fact that the application is in channel selection) is that the number of channels probed is constrained by the cost of probing a channel and there is no possibility of recall.

Zheng [12] has proposed a distributed opportunistic scheduling for ad-hoc communication where links contend for the same channel for transmission. Once the contention is resolved,
the channel condition of the channel is measured. If it is poor, it is assumed that it is better to give up the channel and re-contend for the channel with the expectation that a link with a better channel condition will win the re-contention. They obtain the optimal stopping time [13] to make the selection in order to maximize the overall network throughput. The cost in this problem is measured as the delay that is incurred by rejecting a channel and then to re-contend for a new channel.

Ai [14] propose an optimal stopping relaying strategy in another kind of selection problem, which is selecting the best next hop relay node from a set of candidate nodes in an Ad-hoc network. The overhead to obtain information from the next-hop relay nodes is considered the cost. The stopping rule is very similar to above mentioned Kanodia’s strategy which is to select the node that has a greater reward than a defined threshold value. Again, there is no time constraint or the ability to recall featured in this solution.

There is one other work in the channel probing area proposed by Nicholas and Mingyan [15] where the ability to recall a previously probed channel and allowing selection of a non-probed channel based on the channel distribution is considered. The selection strategy has three actions that can be taken: probe a particular channel from a set of non-probed channel, select the best channel from the list of probed channels or select an non-probed channel by guessing. The cost is for probing a channel is defined based on the parameters like time taken to probe, interference and is assumed to be different for different channels. They consider two constraints for the problem – constant access time and constant data time [16], but there is no ability to recall a previously probed channel.
The work that is closest to our selection problem can be mathematically formulated as a finite memory secretary problem proposed by Smith [17]. It is a modification of the secretary problem strategy [18]. The problem allows the employer to have finite memory of size $m$ that can hold the last $m$ interviewed candidates. This gives the employer freedom to go back to one of the previously interviewed candidates that the employer can remember. This is very similar to the constraint that we have in our problem where the MC can go back to a previously visited MR if it did not find a better MR. However, there is a difference that in our problem the finite memory reduces linearly with visiting more MRs while in their case it remains a constant. This is also true as we impose restrictions on the time-constraint.
Chapter 3

3. Proposed Strategies

In this chapter, we introduce various algorithms we proposed to solve the problem described earlier. We start with the most straight-forward approach followed by three other new strategies.

3.1 Naive Strategy

The most straight-forward strategy for a MC in a WMN is to connect to the first MR that it contacts to get access to the network. This is the fastest way to get access since there is no overhead involved and also does not require any prior knowledge about the network, making it a completely distributed algorithm. However, it might not be the best method since it does not take into account the quality of service provided by that MR. If the MR is already loaded, the MC will receive a poor service for the entire time it is connected to the network. The probability of selecting the best MR is $1/n$ if $n$ is the total number of MRs in the network. This we term as the random or the naïve strategy.

Three different strategies are proposed which can make use of the constraint $k$ (number of MRs that can be visited before making the selection) that has been added to the problem to achieve better quality of service for the new MC that wants to connect to the network. Without loss of generality, $k$ is assumed to be odd to make the analysis simpler.
3.2 Opportunistic strategy

The opportunistic strategy is based on the well known problem, Secretary problem or the Sultans Dowry problem [19]. According to the Secretary problem strategy, the optimal stopping policy to be followed is to skip $n/e$ (where $e$ is the numerical constant of value 2.71828) of the MRs and then select the MR that is better than all the previously visited MRs. If no better MR can be found, then the last MR that is visited is selected. For large $n$, The resulting probability of selecting the best MR in this strategy has been proved to be 0.37. The disadvantage in this strategy is that there is no time limit and this means it can take too long for the MC to find a better MR than all the visited MRs. Also, in the worst case when the best MR lies in the first $n/e$ MRs, then the last MR is chosen irrespective of the quality of service provided by that MR. This means that the MC has to visit all the MRs and choose the last one.

The constraint parameter is introduced into the strategy which ensures that the selection happens after visiting $k$ MRs at the maximum. This changes the strategy of the MC to skip $k/e$ of the MRs and then continue to examine the other MRs until it finds a better MR or settle for the $k^{th}$ MR, whichever occurs first. After this modification, the probability of selecting the best MR is about $0.5k/n$ for small $k$. It is still better than the random strategy.

The Secretary problem strategy is not the optimal strategy with this new constraint. The disadvantage in this strategy is that after skipping the initial $k/e$ MRs, if there is no better MR found, the MC settles for the $k^{th}$ MR. This means there is no guarantee in the selection made following this strategy. The algorithm for the opportunistic strategy is provided in Algorithm 3.1
and the corresponding flow-chart is given in Figure 3.1. An example of the opportunistic strategy is illustrated in Figure 3.2.
Algorithm 3.1 $OpportunisticStop(r_{cv},k)$

1: $cv$; (index representing currently visited MR, set to one at the start of selection process)
2: $cb$; (index representing the current best MR, set to zero at the start of selection process)
3: $f(r_{cb}) = 0$;
4: Calculate the ranking value $f(r_{cv})$;
5: Loop steps 6 through 15 indefinitely
6: if $cv > k/e$ then
7: if ($f(r_{cv}) > f(r_{cb})$ or ($cv == k$)) then
8: Stop examining and choose $cv$.
9: end if
10: else
11: if $f(r_{cv}) > f(r_{cb})$ then
12: Update $f(r_{cb}) \leftarrow f(r_{cv})$ and $cb \leftarrow cv$;
13: end if
14: end if
15: Increment $cv$ by one;
Cv = index which keeps track of the currently visited MR
Cb = index which keeps track of the best MR visited
f(rx) = function that returns the ranking of MR rx with a higher number representing a higher ranking
k = time constraint represented as number of MRs that can be visited

Figure 3.1 Flowchart for the Opportunistic strategy
A graphical representation of the working of the algorithm is given below,

![Graphical representation](image)

- \(n = 9\)
- \(k = 7\)
- \(i = k/e \approx 3\)

Node Selected,
Last node analyzed

**Figure 3.2 Example of the operation of the Opportunistic Strategy**

### 3.3 Conservative strategy

The conservative strategy works in a way opposite to that of the opportunistic strategy. While the opportunistic strategy favors taking risks by moving ahead in search of a better MR, the conservative strategy restrains from taking that risk. In this strategy, the MC visits a subset of MRs, more specifically \(((k+1)/2)\) of the MRs where \(k\) is the time constraint. It remembers the information regarding the best MR that it has encountered over the visited MRs. Let the current best MR be called \(r_{cb}\). The next MR is examined only if the MC can return to \(r_{cb}\) within the constraint \(k\). If \(r_{cb}\) will be lost by examining the next MR, the MC skips examination of further MRs and returns to \(r_{cb}\).

Consider an example with a sample set of \(n\), where \(n = 9\) and the time constraint \(k = 5\). Let the ranks of the \(n\) MRs in consideration be \(\{r_1, r_2, ..., r_9\}\). Let the order in which MC will be visiting the MRs be \(r_1, r_2, ..., r_9\) and let MR \(r_2\) be the highest ranked MR, i.e., \(r_2 = \max_{1 \leq i \leq 9} f(r_i)\). According to the Conservative strategy, MC will visit the MRs until the \((k+1)/2\) MR, i.e., \(r_3\). After visiting \(r_3\), if the MC decides to examine \(r_4\), it will lose the current best MR which is \(r_2\). Therefore, it will return to \(r_2\). If the best MR was \(r_3\) instead, then the MC would have moved to
because it can still come back to $r_3$ if $r_4$ is not better than $r_3$. This is a reasonable strategy in a situation where the time limit is constrained and there is a possibility of returning back to a previously visited MR.

Let $P(r^*|k)$ be the probability that the best MR $r^*$ is selected in $k$ and $P(r^*|m,k)$ be the probability that the MC stops at the $m^{th}$ MR and selects $r^*$, i.e.,

$$m = \frac{k + 1}{2} + x,$$

where $x$ is the number of MRs visited after examining $\frac{k + 1}{2}$ MRs. Then,

$$P(r^*|k) = \sum_{x=0}^{\frac{k+1}{2}-1} P(r^*|x, k). P(x|k).$$

For $k = 5$, the first three MRs are skipped as MC can always come back to the first one. MC will stop at $m = 3$ if $r_1$ or $r_2$ is ranked the highest among the visited three. The probability that $r_1$ or $r_2$ to be the highest ranked MR is $\frac{2}{9}$. Similarly, MC will stop at $m = 4$ if $r_3$ or $r_4$ is the highest ranked MR. As per the above equation,

$$P(r^*|5) = \sum_{x=0}^{1} P(r^*|x, k = 5). P(x|k = 5)$$

$$= P(r^*|x = 0, k = 5). P(x = 0|k = 5) + P(r^*|x = 1, k = 5). P(x = 1|k = 5)$$

$$= P(r^*|m = 3, k = 5). P(m = 3|k = 5) + P(r^*|m = 4, k = 5). P(m = 4|k = 5)$$

$$= \frac{2}{9} + \frac{4}{27}.$$
Calculating this for \( k = 7 \), the same reasoning can be applied with some differences for \( m = 5 \) and \( m = 6 \). For \( m = 5 \), \( r_5 \) is examined only if \( r_3 \) or \( r_4 \) is the highest ranked among the first four, i.e., with a probability of 0.5. For \( m = 6 \), \( r_6 \) is examined only if \( r_3 \) or \( r_4 \) is ranked highest among the first four and \( r_5 \) is ranked higher in the first five MRs, i.e., with a probability of 0.1.

\[
P(r^*|k) = \sum_{x=0}^{2} P(r^*|x, k = 7). P(x|k = 7)
\]

\[
= P(r^*|x = 0, k = 7). P(x = 0|k = 7) + P(r^*|x = 1, k = 7). P(x = 1|k = 7)
+ P(r^*|x = 2, k = 7). P(x = 2|k = 7)
\]

\[
= P(r^*|m = 3, k = 7). P(m = 3|k = 7) + P(r^*|m = 4, k = 7). P(m = 4|k = 7)
+ P(r^*|m = 5, k = 7). P(m = 5|k = 7)
\]

\[
= \frac{2}{9} + \frac{2}{9} + \frac{3}{45}
\]

\[
= \frac{23}{45}.
\]

As seen from the examples, calculating the probability \( P(x|k) \) depends on the outcome of the previous history which makes it difficult to calculate \( P(r^*|k) \) as \( k \) increases. More specifically, when the set of candidate nodes are situated greater than \( (k+1)/2 \), the probability calculation becomes complex because of the inclusion of the previous history. In order to simplify the analysis, an analogy is drawn to the current problem. The analogy removes this odd dependency on the previous history and provides an approximate probability.
Consider a card game consisting of \( n \) cards and two persons A and B playing the game. Each card contains a number in the range \([1, n]\). Let \( k \) be an integer such that \( k < n \). The game is played as follows:

- B draws \( \frac{k+1}{2} \) cards from the \( n \) cards. B can see the numbers on the drawn cards but not A.
- A picks two cards from B’s hand. They are not replaced on the stack.
- If one of the picked card is the highest among B holds, the game stops else B draws one card from the stack and the game continues from the previous step.

This models the same problem as the MR selection problem. The \( n \) cards represent the set of MRs and \( k \) represents the constraint (numbers of MRs that can be visited before making the selection). Initially, B draws \( \frac{k+1}{2} \) cards. This is the step where the MC visits the initial \( \frac{k+1}{2} \) MRs and keeps track of the best MR until that point. When the MC moves to the next MR, it cannot return to the first two MRs that it visited (assuming \( k \) is odd). If the current best MR is the first or second MR, as per the strategy the MC does not examine the next MR. If not, the MC examines the next MR and loses the opportunity to return to the first two MRs. This is the step where A takes two cards and checks if this is the highest among what B holds. If so, the game stops. Otherwise, B picks another card from the stack.

Now, to determine the probability of A choosing the highest card, let \( m \) be defined as the total number of cards picked by B (including the initial set picked) before the game stops.

\[
m = \frac{k+1}{2} + x,
\]
where $x$ is a variable representing the number of cards picked by A from B’s hand. Then, the probability that A picks the card $c^*$ which contains the highest number $n$ is:

$$P(c^*|k) = \sum_{x=0}^{\frac{k+1}{2} - 2} P(r_1 < r^* < r_m|m,k) \cdot P\left(m = \frac{k + 1}{2} + x \mid k\right).$$  

Equation (3.1)

In Equation (3.1), $P(c_1 < c^* < c_m|m,k)$ is probability that card $c^*$ is in the B’s hand which is,

$$P(c_1 < c^* < c_m|m,k) = \frac{\frac{k+1}{2} + x}{n}.$$  

Equation (3.2)

In Equation (3.1), $P(m = \frac{k+1}{2} + x \mid k)$ is the probability that the game stops after A fails for $x$ number of times or equivalently B wins for $x$ number of times. Since $(k+1)/2$ is constant for given $k$, it can be also written as

$$P(X = x|k), \text{ where } X = \{0, 1 \ldots \left(\frac{k+1}{2} - 2\right)\}.$$

$$P(X = x|k) = P(W_{2x+1} > V_{2x+3}, W_{2x-1} < V_{2x+1}, W_{2x-3} < V_{2x-1}, \ldots, W_1 < V_3)$$

where $V_y = \max_{\frac{k+1}{2} + x \leq y \leq \frac{k+1}{2} + x} \{f(r_j)\}$

$$W_z = \max \{f(r_z), f(r_{z+1})\}$$

$$P(X = x|k) = P(W_{2x+1} > V_{2x+3}, W_{2x-1} < V_{2x+1}, W_{2x-3} < V_{2x-1}, \ldots, W_1 < V_3) \ast$$

$$P(W_{2x-1} < V_{2x+3}, W_{2x-3} < V_{2x-1}, \ldots, W_1 < V_3) \ast$$

$$P(W_{2x-3} < V_{2x-1}, W_{2x-5} < V_{2x-3}, \ldots, W_1 < V_3) \ast$$

Equation (3.3)

......
\[ P(W_3 < V_5/W_1 < V_3) * \]
\[ P(W_1 < V_3). \]

Here, \( P(W_1 < V_3) \) represents the probability that A does not picks up the highest card from B when \( X = 0 \), \( P(W_3 < V_5/W_1 < V_3) \) represents the probability that A does not pick up the highest card from B when \( X = 1 \) given that A did not pick up the highest when \( X = 0 \) and so on. At \( X = 0 \), the probability that A does not pick the highest card from B is \((1 - P(W_1 > V_3))\) where \( P(W_1 > V_3) \) is the probability that A picks the highest card from B. \( P(W_1 > V_3) \) is given as \( \left( \frac{2}{k+1} \right) \), i.e. one of the two cards that A picks from the \( \frac{k+1}{2} \) cards is the highest. Therefore,

\[ P(W_1 < V_3) = 1 - \left( \frac{2}{k+1} \right). \]

On similar lines,

\[ P(W_3 < V_5/W_1 < V_3) = 1 - \left( \frac{2}{k+1} \right) \text{ and so on.} \]

In general for \( X = i \), the probability can be given as

\[ P(W_{2i+1} > V_{2i+3}/W_{2i-1} < V_{2i+1},W_{2i-3} < V_{2i-1}, \ldots, W_i < V_3) = \left(1 - \left( \frac{2}{k+1} \right) \right), \quad \text{Equation (3.4)} \]

\[ P(W_{2x+1} > V_{2x+3}/W_{2x-1} < V_{2x+1},W_{2x-3} < V_{2x-1}, \ldots, W_1 < V_3) \] is the probability that A does pick up the highest card from B when \( X = x \). This is given as:
Substituting Equation (3.4) and Equation (3.5) in Equation (3.3),

\[
P(X = x|k) = \frac{2}{k+1+x-2x} \prod_{i=0}^{x-1} \left(1 - \left(\frac{2}{k+1+i-2i}\right)\right)
\]

Expanding the product term and solving, we get the probability to be

\[
P(X = x|k) = \frac{4(k-2x-1)}{(k+1)(k-1)}.
\]  

Equation (3.6)

In Equation (3.1), the range of \( x \) is \( 0 \) to \( \left(\frac{k+1}{2} - 2\right) \) because an MC can never reach the \( k^{th} \) MR. At \( (k-1)^{th} \) MR, depending on the rank of that MR, it will either choose \( (k-1)^{th} \) MR or \( (k-2)^{th} \) MR.

Substituting Equation (3.2) and Equation (3.6) in Equation (3.1), the probability of choosing the highest ranked MR \( r^* \) out of \( n \) MRs and with a constraint of \( k \) is given as:

\[
P(r^*|k) = \frac{2k^2 - k + 9}{3n(k+1)}.
\]  

Equation (3.7)

From Equation (3.7), it can be inferred that an MC can determine the probability of choosing the best MR if the total number of MRs \( n \) and the time constraint \( k \) is known. Likewise, if the MC has a desired probability of being able to choose the best MR and knows the total number of MRs \( n \), it can determine the time constraint \( k \). The algorithm for the conservative strategy is detailed in Algorithm 3.2, the corresponding flow-chart in Figure 3.3 and an example of the conservative strategy is illustrated in Figure 3.4.
Algorithm 3.2 ConservativeStop($r_{cv}, k$)

1: $cv$; (index representing currently visited MR, set to one at the start of selection process)
2: $cb$; (index representing the current best MR, set to zero at the start of selection process)
3: $f(r_{cb}) = 0$;
4: Calculate the ranking value $f(r_{cv})$;
5: Loop steps 6 through 13 infinitely
6: if $(cv + (cv - cb)) < k$ then
7: if $f(r_{cv}) > f(r_{cb})$ then
8: Update $f(r_{cb}) \leftarrow f(r_{cv})$ and $cb \leftarrow cv$;
9: end if
10: Increment $cv$ by one;
11: else
12: Stop examining and select $cb$
13: end if
Cv = 1
Cb = f(r_ab) = 0
k = maximum constraint

Calculate f(r_ab)

Is (Cv + (Cv - Cb)) < k?

Yes

Is f(r_ab) < f(r_ab)?

Yes

f(r_ab) = f(r_ab)
Cb = Cv

No

No

Cv = Cv + 1

Return Cb

Cv = index which keeps track of the currently visited MR
Cb = index which keeps track of the best MR visited
f(r_ab) = function that returns the ranking of MR r_ab with a higher number representing a higher ranking
k = time constraint represented as number of MRs that can be visited

Figure 3.3 Flowchart for the Conservative strategy
3.4 Calculatingly-Opportunistic strategy

The opportunistic strategy fails since it does not take into consideration the ability to return to a previously visited MR. So, a hybrid strategy is proposed to take advantage of both these algorithms. In this strategy, the MC starts with the opportunistic strategy by skipping the first \( (k/e) \) MRs and proceeds to find a MR which is better than the previously visited MRs. Once it finds such an MR, it follows the conservative strategy, i.e., it moves to the next MR, only if it can still return to the current best MR. In this way, the strategy ensures that the MC does not lose out on finding out a better ranked MR by not utilizing the time constraint completely. The algorithm for this strategy is detailed in Algorithm 3.3, the flow-chart in Figure 3.5 and an example of the calculatingly-opportunistic strategy is illustrated in Figure 3.6.
Algorithm 3.3 \textit{CalOpportunStop}(r_{cv}, k)

1: \texttt{cv}; (index representing currently visited MR, set to one at the start of selection process)
2: \texttt{cb}; (index representing the current best MR, set to zero at the start of selection process)
3: \texttt{f(r_{cb}) = 0};
4: Calculate the ranking value \texttt{f(r_{cv})};
5: \textbf{if} \texttt{cv < k/e} \textbf{then}
6: \hspace{1em} \textbf{if} \texttt{f(r_{cv}) > f(r_{cb})} \textbf{then}
7: \hspace{2em} \text{Update} \texttt{f(r_{cb}) ← f(r_{cv})} \text{ and} \texttt{cb ← cv};
8: \hspace{1em} \textbf{end if}
9: \textbf{else}
10: \hspace{1em} \textbf{if} \texttt{f(r_{cv}) > f(r_{cb})} \textbf{then}
11: \hspace{2em} \text{Call} \textit{ConservativeStop}(r_{cv}, k);
12: \hspace{1em} \textbf{end if}
13: \textbf{end if}
14: \texttt{Increment cv by one;}
Figure 3.5 Flowchart for the Calculatingly-Oppportunistic strategy
A graphical representation of the working of the algorithm is shown below,

\[ n = 9 \]
\[ k = 7 \]
\[ i = \frac{k}{e} \approx 3 \]

Figure 3.6 Example of the operation of the Calculatingly-Opportunistic Strategy
Chapter 4

4. Simulation Results

This chapter focuses on the evaluation part of the research. The experimental analysis of the proposed strategies are done in two methods:

- **Numerical Simulation**: A simulation to study the performance of the different strategies and to verify the theoretical result obtained for the conservative strategy that has derived in Chapter 3.

- **Network Simulation**: A study of the strategies on a network using the Qualnet network simulator [20].

4.1 Numerical Analysis

The three strategies proposed are evaluated using numerical simulations to determine which is the most efficient in finding the best MR from a set of MRs if they are linearly probed. Two different numerical experiments are run to study the algorithms:

- Probability of selecting the best ranked MR \( r^* \).

- Probability of selecting one of the 3 best ranked MRs.

4.1.1 Numerical Simulation Setup

This simulation is run in Java [21]. A sample set of 100 numbers are randomly generated. These numbers can be compared to the MR’s ranking parameter that will be assigned by the MC
when it visits each MR. Each number is an 8-digit floating point number uniformly selected in the range \([0, 1]\). This is intended to ensure that all the values are unique. On each sample, all the three strategies are run for different values of \(k\) and the probability of success for each strategy is calculated. Depending on the type of experiment, when a strategy picks the maximum or one of the top three maximum (of the 100 numbers), it is termed as a success.

### 4.1.2 Numerical Simulation Results

The results of the experiment are shown in Figure 4.1 and Figure 4.2.
4.1.3 Inference from the results

From Figure 4.1, it can be observed that the conservative strategy is the most effective among the three strategies. The conservative strategy beats the opportunistic strategy by a huge margin. For instance, to achieve even a probability of 0.1, the opportunistic strategy requires twice the constraint ($k$) that is required for the conservative strategy. The calculatingly-opportunistic strategy has almost the same efficiency as the conservative strategy for smaller values of $k$ but with increase of $k$, it also loses out to the conservative strategy.
Figure 4.2 shows the probability for selecting one of the top 3 ranked MRs. It is not necessary for a MC to always select the best MR. It could benefit by choosing one of the top 3 ranked MRs. The result again proves that the conservative strategy wins over both the other strategies.

The results can be summarized as follows:

- Conservative strategy wins over both opportunistic and calculatingly-opportunistic strategy. For very small values of $k$, conservative and calculatingly-opportunistic strategies perform almost the same. Therefore, it is more likely that both the strategies result in choosing the same MR most of the times.

- The plot for the theoretical probability of the conservative strategy is almost a perfect fit to the plot for the probability obtained for the conservative strategy through numerical simulation. This is additional proof that the approximation made during the derivation was a reasonable one.

4.2 Ranking parameter

While multiple algorithms have been proposed for selecting the best MR, it is important to define on what basis one MR is better than another. The parameter that is used to define this is called the ranking parameter and the choice of this parameter affects how the algorithm performs. If a wrong parameter is used to rank the MRs, it is very much possible to end up with undesired results. There are multiple variables in a network that can be used as the ranking parameter. We list a few of the parameters and discuss whether they can be chosen as the ranking parameter:
- **Shortest path**: A path is defined as the collection of MRs traced by a packet when it is sent from a MR to the IGW. The shortest path is the path that has the minimum number of MRs to reach the IGW.

- **RTS retransmissions (MAC)**: This defines the number of RTS messages that were retransmitted on each link as part of the CSMA protocol. This value provides an indication of the channel contention level and the quality of a link [22].

- **Packet retransmissions (MAC)**: This defines the number of packet retransmissions at the MAC layer. In 802.11, a packet is retransmitted at the MAC level when there is no acknowledgement received for the sent packet [23]. This is assumed as an error in the transmission and the packet is retransmitted.

- **Packet drops (Network)**: This defines the number of packet that were dropped at a node. This would be mainly due to the reason that the queue/buffer was full. This indicates the load on that node.

The above mentioned parameters capture quality of the network at different layers in the network. The shortest path might be a good parameter if the links in the network are all equally burdened, but unfortunately this is not the case in a WMN. The RTS and packet retransmissions parameters are a good indication of the quality of the link between two MRs, but not in terms of the load on the link. The packet drops parameter represents the load an MR and provide an idea whether the MR can handle extra traffic or not. Therefore, we choose this parameter as the ranking parameter.
4.3 Network Simulation

4.3.1 Qualnet

Qualnet is an event-driven network simulator. It provides an environment for creating new network protocols, visualizing network scenarios and evaluating the performance of such networks [20]. It provides both graphical and command-line interfaces for running simulations. There are several advantages of using Qualnet of which some are highlighted below:

- availability of both graphical and command-line interface,
- availability of various standardized protocols within the simulator,
- ability to run batch simulations,
- ease of integration with external programs (eg. Java and perl), and
- detailed and organized output data for simulations.

More information on Qualnet and its usage can be found in [20]. Qualnet version 4.5 is used in the simulations.

4.3.2 Experimental Setup

While Qualnet was the core engine used for the simulations, the experimental setup included several more modules. Figure 4.3 shows the overall setup.
- Simulation Controller: This is the central module that controls the entire simulation. It controls the different modules, the time of invocation and the order of execution of the other modules.

- Topology Generator: This module generates the network topology for each simulation, more specifically a grid topology. It generates a qualnet understandable file which identifies the positioning of the nodes in a network given the number of nodes and the distance between the nodes.

- Traffic Generator: This module generates a qualnet understandable file which lists the nodes which generate traffic in the network along with the type and amount of traffic for that simulation.

- Channel Allocator: This module assigns the channels to the interfaces of each node in the network for each simulation.
- Network Configuration Generator: This module creates a qualnet understandable file that sets the different parameters that defines the configuration of the network.

- Algorithm Processor: This module runs the different algorithms/strategies which defines the selection process. It performs the ranking of the MRs based on the chosen ranking parameter and selects the best MR considering the time constraint (k) that is defined for the simulation.

- Qualnet Engine: The Qualnet engine is invoked by the Simulation Controller to run the simulation after the network is configured. There are several input files for Qualnet which are generated in the different modules described here.

- Statistics Parser: This module executes at the end of the simulation to process the output of the simulation and generate meaningful results from the data.

Figure 4.4 shows the flow of execution in the experimental setup.

![Figure 4.4 Execution Flow](image)

The simulation uses a recursive feedback approach. Since the MC has to visit the MRs and rank them before making the selection, the simulation is run initially to obtain the ranking of the
MRs followed by the application of the various algorithms. Then, the simulation is run again with the exact configuration as before but with the new MC attached to the selected MR.

### 4.3.3 Simulation Setup

This section describes the network configuration that was used for the simulations. The simulations were broadly classified into two different configurations:

- A grid topology of 81 nodes in a square area of 2000m X 2000m (as shown in Figure 4.5).
- A random topology of 60 nodes in a square area of 1000m X 1000m (as shown in Figure 4.6).

The radio range is adjusted such a way that all adjacent nodes are connected. We assume that there is only one IGW in the network. The IGW is located at the approximate centre of the network. The traffic flows from the boundary nodes towards the IGW. Therefore, the boundary nodes are assigned a constant bit rate (CBR) traffic generator. The MAC protocol used is CSMA/CA [22] with RTS/CTS and ACK enabled. The mobility model allows a MC to randomly enter from the boundary of the network and move along the boundary of the network. Also, it is assumed that an MC can connect to only one MR at any time.

Under each topology, the strategies are applied for the network with:

- Two different routing protocols, a dynamic (Adhoc On-Demand Distance Vector [24]) and a static (Bellman-Ford [25]) routing protocol.
- Five different values for the constraint $k$ ($k = \{5, 6, 7, 8, 9\}$).
- Three different traffic rates ($CBR = \{256, 341, 512\}$ bytes/s).
Each simulation is run for 30s and the results are averaged over multiple runs (20) with different seed values.

Figure 4.5 Snapshot of a WMN in Grid topology
Figure 4.6 Snapshot of a WMN in Random topology
4.3.4 Simulation Results

From the simulation, the following graphs are obtained:

- **Network Throughput vs constraint k**: We track the overall throughput of the network before a new MC joins the network and after a new MC joins according to each of the strategy. Therefore, this graph has the following plots – base (network throughput before any change in the traffic condition), random (network throughput when the new MC joins according to the naïve strategy), opportunistic (network throughput when the new MC joins following the opportunistic strategy), conservative (network throughput when the new MC joins following the conservative strategy). This graph gives a comparison of how much network throughput can be gained by following each of the strategies.

- **New MC throughput vs constraint k**: This is similar to the graph above but here we consider the throughput of the new MC. The throughput of the new MC is related to the MR that it selected following the strategy. The following plots are part of this graph – random, opportunistic and conservative.

- **Success rate**: We also calculate the success rate of each of the strategies. The success rate is defined as the percentage of times that a MR selected according to a strategy helped in bettering the throughput of the network or throughput of the new MC.
4.3.4.1 Results for 9 x 9 Grid Network Topology

Figure 4.7 Results for AODV, traffic rate = 256 bytes/s
Figure 4.8 Results for AODV, traffic rate = 341 bytes/s
Figure 4.9 Results for AODV, traffic rate = 512 bytes/s
Figure 4.10 Results for Bellman-Ford, traffic rate = 256 bytes/s
Figure 4.11 Results for Bellman-Ford, traffic rate = 341 bytes/s
Figure 4.12 Results for Bellman-Ford, traffic rate = 512 bytes/s
4.3.4.2 Results for 9 x 9 Random Topology

Figure 4.13 Results for AODV, traffic rate = 256 bytes/s
Figure 4.14 Results for AODV, traffic rate = 512 bytes/s
Figure 4.15 Results for Bellman-Ford, traffic rate = 256 bytes/s
Figure 4.16 Results for Bellman-Ford, traffic rate = 512 bytes/s
4.3.5 Inference from the Results

The following are the inferences from the results obtained,

- The conservative strategy proves to be the best of all the strategies evaluated. The application of conservative strategy improves both the MC’s throughput as well as the overall network throughput.

- The results are positive for the conservative strategy irrespective of the routing protocol used (Bellman-Ford or AODV). This shows that the conservative strategy works independent of the routing protocol used in the network.

- Increase in the traffic load on the MRs has not resulted in a change in the behavior of the conservative strategy. The throughput has also gracefully increased which shows that the strategies are applicable to networks with heavy traffic load.

- The throughput gain at the network level is relatively small because of the size of the network and considering that there is only one traffic source newly added into the network. In comparison, the throughput gain at the MR level is significant. The maximum gain obtained is 18% and on an average the throughput gain is 8%, which for a MC is a significant improvement.

- The opportunistic and the random strategy also have been able to improve the throughput in some scenarios, but the results have not been consistent. This is because of the fact that there is a risk factor involved in the selection process in the two strategies. While for conservative strategy, the results are consistent since there is an assurance that the MR chosen is the best in the set of candidate MRs that a MC can visit.
- In certain scenarios, the throughput (overall network or new MC) obtained is equal for two or all the strategies. This can happen when the strategies result in selecting the same MR.

- From the success percentage obtained for each of the simulations in the grid topology, it is clear that following the conservative strategy results in higher probability of bettering the network throughput. More specifically, an MR selected using the conservative strategy has more than 55% chance of improving the network and the new MC’s throughput over the random or opportunistic strategy whose success percentages are 26% and 19% respectively.
Chapter 5

5. Conclusion and Future work

5.1 Conclusion

We propose and analyze three different strategies that can be applied in a time-constrained MR selection problem. The mathematical and statistical analysis show that the application of conservative strategy results in a higher probability of choosing the best ranked MR as compared to the opportunistic or the calculatingly-opportunistic strategy. We also identify the network parameter to be used by the MC in ranking the MRs. The selection of this ranking parameter is crucial for the effectiveness of the algorithm. We apply the opportunistic and conservative strategy on different network configurations using a network simulator to evaluate their performance. The simulations show that the conservative strategy not only helps in achieving better throughput for the individual MC, but also in bettering the throughput of the overall network. Also, we find that the conservative strategy has a higher success rate compared to the other strategies.
5.2 Future Work

The work presented in this thesis allows for many variations in the scenario that can make the problem even more interesting. A few of these are listed below,

- While we consider only one MC entering the network at a given point of time, the strategy can be applied in a network where multiple clients enter simultaneously. We predict that this would result in a much higher increase in the throughput for the network.

- Temporal variations in the ranking parameter can be added to the problem to make the problem more realistic, i.e., assuming the ranking values changes over time.

- Considering a combination of network parameters instead of a single network parameter can allow for focusing on multiple important aspects of a network rather than just one. Real-time applications like audio/video applications can look for more bandwidth rather than reliability while for file transfer applications the importance is on reliability.
Bibliography


