Analysis and design of quality link metrics for routing protocols in Wireless Networks

Nadeem Javaid

To cite this version:

Nadeem Javaid. Analysis and design of quality link metrics for routing protocols in Wireless Networks. Other [cs.OH]. Université Paris-Est, 2010. English. <NNT : 2010PEST1028>. <tel-00587765>

HAL Id: tel-00587765

https://pastel.archives-ouvertes.fr/tel-00587765

Submitted on 21 Apr 2011

UNIVERSITÉ PARIS-EST

ECOLE DOCTORALE: MSTIC

Thèse de Doctorat

Spécialité : Informatique

Nadeem JAVAID

Analyse et conception des métriques de qualité de liens et routage dans les réseaux sans fil

Thèse dirigée par : Karim DJOUANI

Soutenue le Décembre 15, 2010

Jury :

B.J. VAN WYK	Prof. Tshwane University of Technology - South Africa	**Rapporteur**
J. BEN-OTHMAN	MDC (HDR), Université de Versailles - France	**Rapporteur**
K. BARKAOUI	Prof. CNAM, Paris - France	**Examinateur**
O. BRUN	Directeur de recherche, LAAS-CNRS, Toulouse - France	**Examinateur**
K. DJOUANI	Prof. Université Paris Est - France	**Directeur de thèse**

Laboratoire Images, Signaux et Systèmes Intelligents – EA 3956

Abstract

This dissertation endeavors to contribute enhancements in goodputs of the IEEE 802.11-based Wireless Multi-hop Networks (WMhNs). By performing exhaustive simulations, for the deep analysis and detailed assessment of both reactive (AODV, DSR, DYMO) and proactive (DSDV, FSR, OLSR) protocols for varying mobilities, speeds, network loads and scalabilities, it is observed that a routing link metric is a significant component of a routing protocol. In addition to finding all available paths, the fastest end-to-end route is selected by a link metric for the routing protocol. This study aims the quality routing. In the class of quality link metrics, Expected Transmission Count (ETX) is extensively used. Thus, the most recently proposed ETX-based metrics have been analyzed. Though, newly developed metrics over perform ETX but still they can be improved.

After going through profound analysis and particularized comparison of routing protocols depending upon their classes (reactive and proactive) and ETX-based metrics, we come to realize that users always demand proficient networks. In fact, WMhNs are facing several troubles which they expect to be resolved by the routing protocol operating them. Consequently, the protocol depends upon the link metric for providing quality paths. So, we identify and analyze the requirements to design a new routing link metric for WMhNs. Because, considering these requirements, when a link metric is proposed, then: firstly, both the design and implementation of the link metric with a routing protocol become easy. Secondly, the underlying network issues can easily be tackled. Thirdly, an appreciable performance of the network is guaranteed.

Keeping in view the issues of WMhNs, increasing demands of users and capabilities of routing protocols, we propose and implement a new quality link metric, Interference and Bandwidth Adjusted ETX (IBETX). As, MAC layer affects the link performance and consequently the route quality, the metric therefore, tackles the issue by achieving twofold MAC-awareness. Firstly, interference is calculated using cross-layered approach by sending probes to MAC layer. Secondly, the nominal bit rate information is provided to all nodes in the same contention domain by considering the bandwidth sharing mechanism of 802.11. Like ETX, our metric also calculates link delivery ratios that directly affect throughput and selects those routes that bypass dense regions in the network. Simulation results by NS-2 show that IBETX gives 19% higher throughput than ETX and 10% higher than Expected Throughput (ETP). Our metric also succeeds to reduce average end-to-end delay up to 16% less than Expected Link Performance (ELP) and 24% less than ETX.

Key Words

Wireless multi-hop networks, reactive, proactive, routing protocols, mobility, scalability, traffic, throughput, end-to-end delay, normalized routing load, quality link metric, ETX, inverseETX, IBETX

Resumé

Les travaux de recherche menés dans le cadre de cette thèse concernent l'amélioration du débit et de la qualité de service dans les réseaux sans fil basés sur les standards de la famille 802.11. Des simulations exhaustives ont été menées pour l'analyse et l'évaluation des performances des protocoles de routages réactifs AODV, DSR et DYMO ainsi que des protocoles de routage pro-actifs DSDV, FSR et OLSR. Plusieurs paramètres ont été considérés comme la mobilité des noeuds la charge du réseau et la mise à l'échelle. Nous pouvons observer que la métrique de qualité de lien est un paramètre important dans toute stratégie de routage. L'objectif du présent travail est de proposer une analyse comparative des différents protocoles de routage, basée sur les métriques de qualité de lien et de proposer une nouvelle métrique permettant d'améliorer le routage en termes de routage et de délai bout-en-bout. Dans la disparité des métriques proposées dans la littéraire, la métrique ETX (Expected Transmission Count) a été largement utilisée. Aussi, nous focaliserons, principalement, sur les métriques basées sur ETX. Notre constatation est que les besoins en termes de qualité de service dans les réseaux sans fil multi-saut, sont affecté entre autre par le protocole de routage adopté et la métrique de lien utilisée. Cependant, ceci est aussi strictement lié à la qualité des canaux de communication au niveau physique et l'état des files d'attente au niveau MAC. Aussi, une nouvelle métrique de qualité de lien est proposée, basée sur ETX et appelée IBETX (Interference and Bandwidth Adjusted ETX). Cette nouvelle métrique prend en considération les effets de l'interférence et de l'impact de l'état de la couche MAC sur les performances des liens de communication.

Des simulations ont été conduites sous NS-2 afin de montrer l'intérêt de la métrique utilisée. Ainsi IBETX améliore le débit efficace de 19% par rapport á ETX et de 10% par rapport á ETP (Expected Throughput). Par ailleurs, la métrique proposée réduit le délai bout en bout de 16% par rapport á ELP (Expected Link Performance) et de 24% par á ETX.

Mots Clés

Réseaux sans fil multi-saut, réactif, proactif, protocoles de routage, mobilité, scalibilité, vitesse, charge de trafic, débit, délai de bout-en-bout, normalisé de routage de charge, lien de qualité métrique, ETX, inverseETX, IBETX

Contents

Chapter 1

Introduction

Recent wireless networks are rapidly moving from infrastructured to the infrastructures technologies. Among currently existing communication technologies, the Wireless Multihop Networks (WMhNs) are supposed to be the forerunner. They provide users with flexible structures, feasible cost, convenience, ever growing bandwidths, and innovative solutions. In such networks, wireless nodes are not always in the transmission range of each other. So, the intermediate nodes act as routers to receive and send the routing and data packets from and to the nodes in their transmission range. The increasing demands of the users provide a challenge to improve the quality of WMhNs. These requirements include; continuously increasing speeds of mobile users, randomly varying mobilities, dynamically changing scalabilities of wireless devices, and different data traffic rates.

To achieve an appreciable performance from an underlying WMhN, along with other protocols, routing protocols play a vital role, because they are responsible to create, maintain and synchronize the routing information tables at all wireless nodes in the presence of varying speeds, mobilities, scalabilities and transmission rates. A routing protocol guarantees optimized performance, if it assures efficient routing and Quality of Service (QoS) for WMhNs by enhancing per-flow bandwidth that is achieved by the respective routing link metric. A quality link metric helps its operating protocol to do quality routing by selecting the fastest end-to-end paths. It can better work if it encounters the issues of wireless links where nodes in the same contention domain produce interference and share the common bandwidth. A link metric operating at routing layer can result in outstanding performance, if it takes the relevant information from the MAC layer.

We start from precise evaluation and comprehensive comparison of three reactive protocols: Ah-hoc On-demand Distance Vector (AODV) [1]- [12], Dynamic Source Routing (DSR) [3]- [6], [11]- [12] DYnamic Manet On-demand (DYMO) [21] and three proactive protocols: Destination-Sequenced Distance Vector (DSDV) [13], [14], Fish-eye State Routing (FSR) [24], Optimized Link State Routing (OLSR) [26] in WMhNs. The study evaluates the impact of different mobilities, speeds, scalabilities, and traffic loads using the performance parameters throughput, end-to-end delay, packet loss fraction and normalized routing load. Based upon the extensive simulation results in NS-2, all of six protocols are ranked according to these performance parameters. Besides providing the interesting facts regarding the response of each protocol, the trade-offs made by protocols to achieve the required or desired performances are also studied. Such as, to achieve throughput in a static situation, a protocol has to pay some cost in the form of increased end-to-end delay and in the scenario of moving nodes, cost of routing overhead is to be paid to achieve raise in the number of delivered packets. How, route discovery, route maintenance, and route table calculation processes have been implemented and effect the performance of respective protocols has been discussed with the help of generalized flow charts of both reactive and proactive classes of routing protocols.

Being most popular and IETF standard metric, minimum hop count is appropriately used by Ad-hoc Networks, as new paths must rapidly be found in the situations where quality paths could not be found in due time due to high node mobility. There always has been a tradeoff between throughput and energy consumption, but stationary topology of Wireless Mesh Networks (WMNs) and high node density of Wireless Sensor Networks (WSNs) benefit the algorithms to consider quality-aware routing to choose the best routes from source to destination. We analytically review the ongoing research on wireless routing link metrics which are based on ETX (Expected Transmission Count). The reason to study ETX is that it performs better than minimum hop count metric under link availability. Performances over ETX, target platforms and design requirements of these ETX based metrics are high-lighted. Consequences of the criteria being adopted (in addition to expected link layer transmissions and retransmissions) in the form of incremental: (1) performance overheads and computational complexity causing inefficient use of network resources and instability of the routing algorithm, (2) throughput gains achieved with better utilization of wireless medium resources have been elaborated.

As mentioned above, users demand the best performance from the underlying network, which expects its operating routing protocol to provide quality paths with increased per-link bandwidth by resolving the respective issues: interference, bandwidth sharing, etc. For QoS routing, a protocol heavily depends upon link metric. So, we identify and analyze the requirements to design a new routing link metric for WMhNs. By keeping these design requirements in view if a link metric is proposed, then to design and implement a new link metric for a routing protocol become easy, the target network's issues are easily tackled, and an appreciable performance of the network is guaranteed. Along with the existing implementation of three link metrics, ETX, Minimum Delay (MD), and Minimum Loss (ML), we implement inverse ETX; *invETX* with OLSR using NS-2.34. The simulation results show that how the computational burden of a metric degrades the performance of the respective protocol and how a metric has to trade-off between different performance parameters.

Based upon the requirements to design a new routing metric, we propose a new quality link metric, interference and bandwidth adjusted ETX (IBETX) for WMhNs. Our metric is three dimentional, because it tackles three severe issues existed in WMhNs.

(1) ETX over performs Minimum Hop-count metric by measuring the *asymmetry of links* by calculating the probe delivery ratios in both forward and reverse directions.

Usually, MAC layer degrades the performance of the IEEE 802.11-based networks that consequently effects the transmissions going-on at the higher layer. So, IBETX deeply looks inside the phenomenon and succeeds to achieve MAC-awareness in two ways:

(2) it calculates the inter-flow interference by the aid of cross-layered approach (sending probes to MAC layer).

(3) it provides all the nodes in the same contention domain with the information of nominal bit rate by considering the bandwidth sharing mechanism of 802.11.

Simulations performed in NS-2 reveal that IBETX gives 19% higher throughput than ETX and 10% higher packet delivery ratios than Expected Throughput (ETP), and 10% higher throughput than Expected Link Performance (ELP). Per-link decreased end-to-end delay saves network bandwidth, our metric, therefore succeeds to lower overall delay up to 16% less than ELP, 24% less than ETX, and 15% lower than ETP.

At the next page, Fig.1.1 demonstrates the work done during the completion of this thesis.

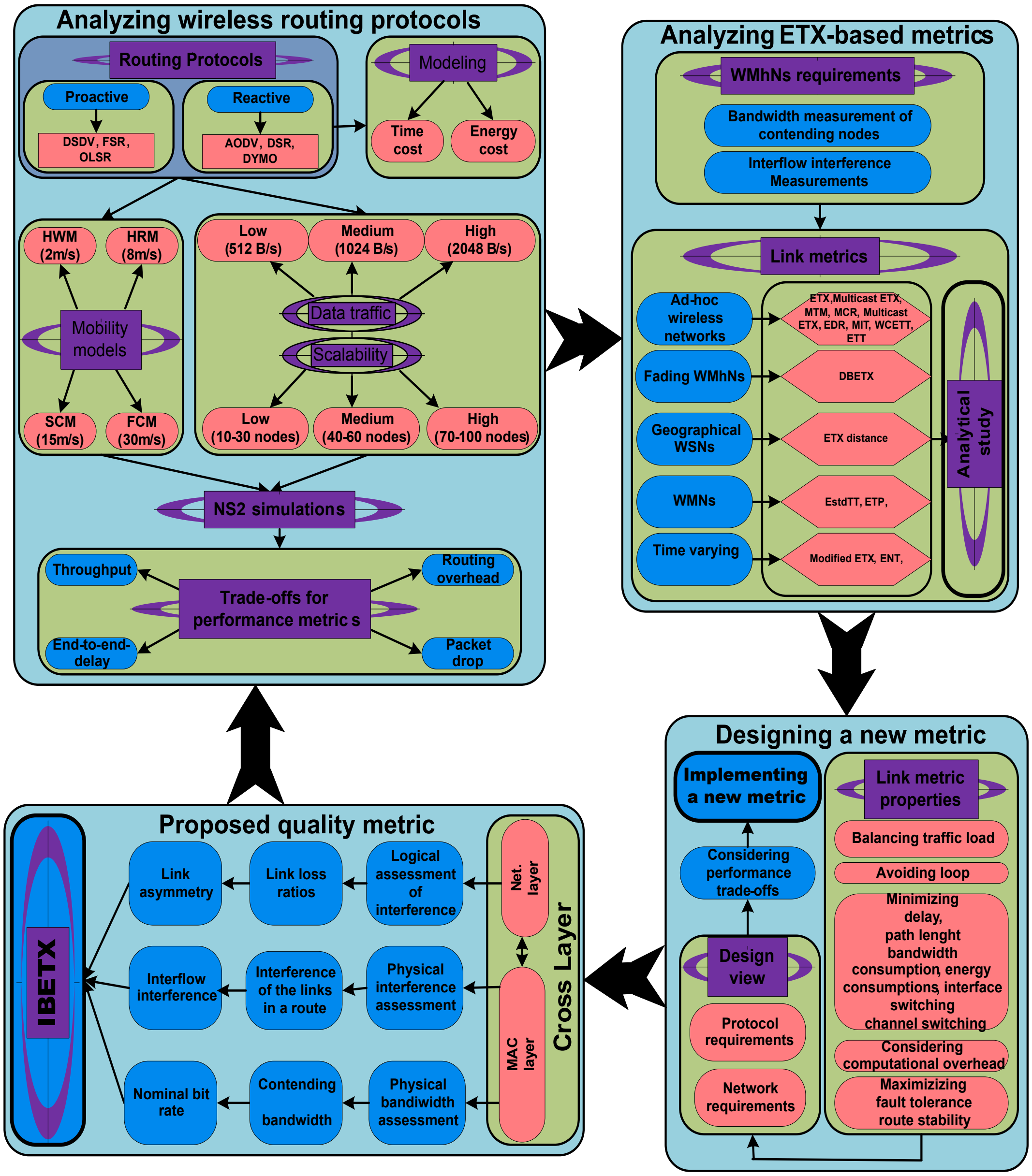


Figure 1.1: Thesis in brief

Chapter 2

Impact of Mobility and Speed

Contents

2.1 Introduction

Wireless communication, in the recent years, has become significantly important, especially when roaming around. The Wireless Multi-hop Networks (WMhNs) provide users with the facility to communicate while moving with whatever the locations and the velocities they want, without disruption. So, mobility plays a crucial role for routing protocols in multi-hop wireless networks. Now, depending upon the desires and requirements in daily life, wireless devices in common use, move with different velocities and in random directions.

To correctly illustrate the performance evaluation of the routing protocols, it is remarkably significant to exactly depict the movement of mobile nodes. So, Shams *et al.* in [15] designed scenario-based mobility models which closely present the movement patterns of users in real life and they have evaluated two reactive routing protocols, AODV and DSDV. The proposed mobility models are: Fast Car Model (FCM), Slow Car Model (SCM), Human Running Model (HRM) and Human Walking Model (HWM). We follow the same models for this study. FCM states that the mobile nodes are vehicles moving up to the speeds of 30m/s or 108km/h [16] on highways and motor ways. In practice, vehicles do not move with this speed all the time rather they take pauses at different break points and traffic signals. Thus, ’pause-time’ intervals are also considered. Like FCM, SCM also considers the vehicles but moving with the speed of 15m/s or 45km/h on the busy roads and cannot move at higher speeds. It is observed that most of the times, wireless devices are carried by the humans. For example, soldiers in the combat zone can run or walk, people jogging on different tracks, in emergency situations, sports and so on. In short, 8m/s or 28.8km/h can be taken as an average speed for SCM. The HWM is identical to the HRM model but with an average speed of 2m/s or 7.2km/h [16]. The examples for HWM may be people walking in the shopping centers, university or college campuses, etc.

Being an interface between the underlying wireless network and mobile users, a routing protocol plays an important role. So, to provide the reader with a comprehensive idea about routing and how do the routing protocols react to the topological changes, we have chosen the most widely experimented and frequently used protocols for our study; three from reactive or on-demand class: Ad-hoc On-demand Distance Vector (AODV), Dynamic Source Routing (DSR), DYnamic MANET On-demand (DYMO), and three from proactive or table-driven class Destination Sequenced Distance Vector (DSDV), Fish-eye State Routing (FSR), Optimized Link State Routing (OLSR). Authors in [15], have analysed two protocols; AODV and DSDV. Simulations are run for four pause times (0s, 1s, 10s and 450s). However, routing protocols being categorized in reactive and proactive classes are yet to be analyzed. Moreover, to perform a precise and detailed analysis we have simulated six protocols with ten pause times (0s, 100s, 200s,. . . and 900s).

2.2 Routing protocols and mobility

This section is devoted to short description of each protocol, stating the routing technique working behind it, class to which the protocol belongs; i. e., reactive or proactive, the way in which it performs *route discovery (RD)*, *route maintenance (RM)*, *route table (RT) calculation* and at the end, the claims made by each protocol to deal with the mobility. At the end of section, Table.2.1 summarizes all of the six protocols.

2.2.1 Reactive protocols and mobility

AODV [17], [18], DSR [17], [19], [20] and DYMO [21], [22] are multi-hop on-demand routing protocols. Their on-demand nature has a great impact on mobility because they compute routes only when needed making them suitable for mobile scenarios. AODV claims that ”it can handle low, moderate, and relatively high mobility rates, as well as a variety of data traffic levels” [18]. DSR claims that ”it adapts quickly to the topological changes when movement of nodes is frequent. It requires little or no routing overhead during the periods in which nodes move less frequently or remain at rest” [20]. DYMO states that “it adapts to changing network topology and determines unicast routes between nodes within the network in ’on-demand’ fashion” [21]. So, in this study we evaluate and compare the performance of these protocols based upon their claims regarding mobility.

AODV alters the basic distance vector algorithm by adding sequence number while DSR and DYMO use source routing algorithm. Distance vector dissemination takes less bandwidth as compared to source routing. All of three protocols use flooding based *RD*for path calculation, as shown in Fig.2.1. AODV uses hop-by-hop routing while DSR and DYMO use source routing as packet forwarding scheme. These protocols implement two common operations: *RD* and *RM*. The *Gratuitous Route Reply (grat. RREP)* strategy is used for optimizations during *RD* process and is used by both AODV and DSR. This process reduces the flooding overhead and helps to find quick routes during mobility. DYMO

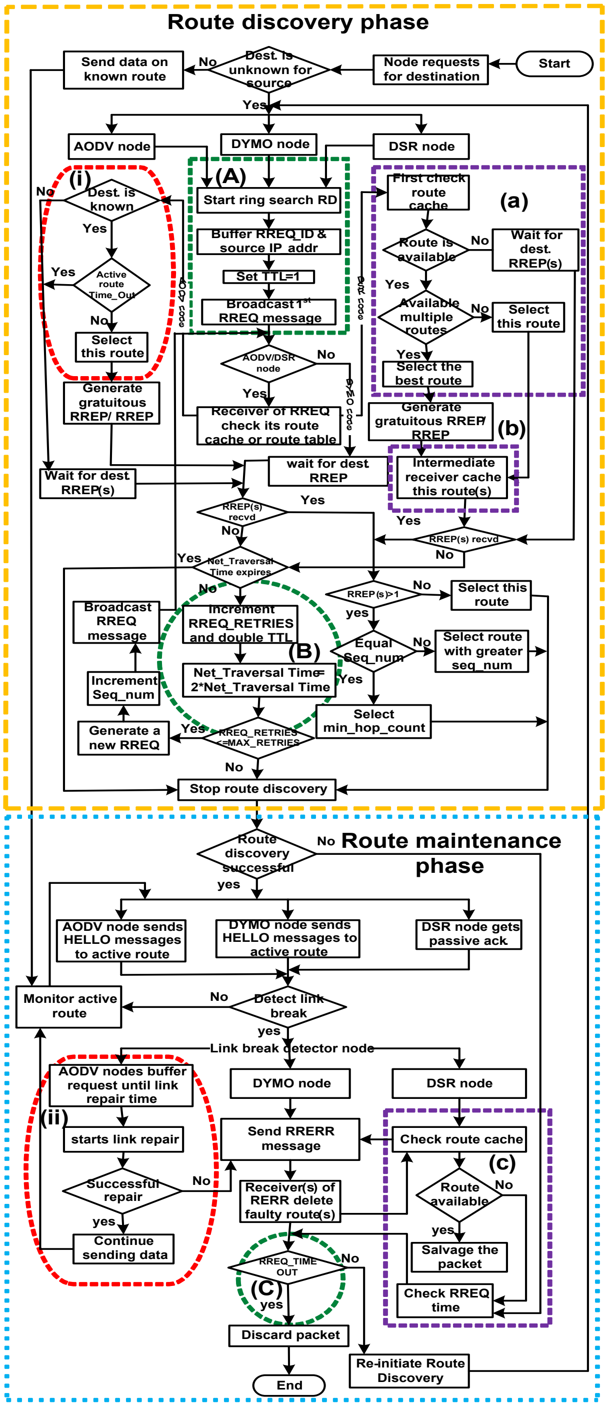


Figure 2.1: Reactive protocols, 'route discovery' and 'route maintenance'.

inherits some features from DSR while most of the features from AODV; that is why it is known as the successor of AODV. Both of the algorithms: *expanding ring search (ERS)* and *exponential back-off (EB)* are used by DYMO in the same manner as DSR and AODV.

AODV uses *local link repair (LLR) mechanism* and DSR uses *packet salvaging (PS) technique*. The previous mechanism helps to repair the links which are broken due to varying topology. Optimization for *RD* is achieved through the use of *route cache (RC)*: the distinguished feature of DSR among reactive protocols. For monitoring the active routes, HELLO messages are used to check the connectivity in AODV and DYMO allowing the mobile nodes to quickly obtain the routes for new destinations that does not require nodes to maintain routes to destinations that are not in active communication. On the other hand, DSR monitors the *active routes* by link layer per hop acknowledgments or through passive acknowledgments.

2.2.2 Proactive Protocols and Mobility

DSDV [23], FSR [24], [25] and OLSR [26], [27], are table-driven proactive protocols. All of these proactive protocols use *hop-by-hop routing* scheme for packet forwarding. In DSDV, distance vector packets are dispersed and then *Distributed Bellman Ford (DBF)* algorithm is used for path calculation, as shown in Fig.2.2. In FSR, DBF algorithm is used for path calculation and link state packets are not flooded. The nodes maintain a link state table based on up-to-date information received from the neighboring nodes and they periodically exchange it with their local neighbors only. For the path calculation OLSR uses *Dijkstra's algorithm*.

To maintain consistency in routing tables, DSDV generates periodic updates, $P_{updates}$ and trigger updates, $T_{updates}$, when information about new links becomes available. For convergence, routing information is advertised by broadcasting the packets periodically. FSR uses *graded-frequency (GF)* mechanism to achieve route accuracy while *Multi-point Relay (MPR)* redundancy mechanism is used by OLSR in high dynamic situations. In DSDV, on request for a destination, data is kept for a duration between arrival of the first and arrival of the best route for each destination. So, decision is made to delay advertising routes which are about to change soon thus minimizing the damping fluctuations of the route tables. That is, advertisements for a particular route which is not stabilized so far are delayed to lessen the number of rebroadcasts of possible route entries that usually arrive

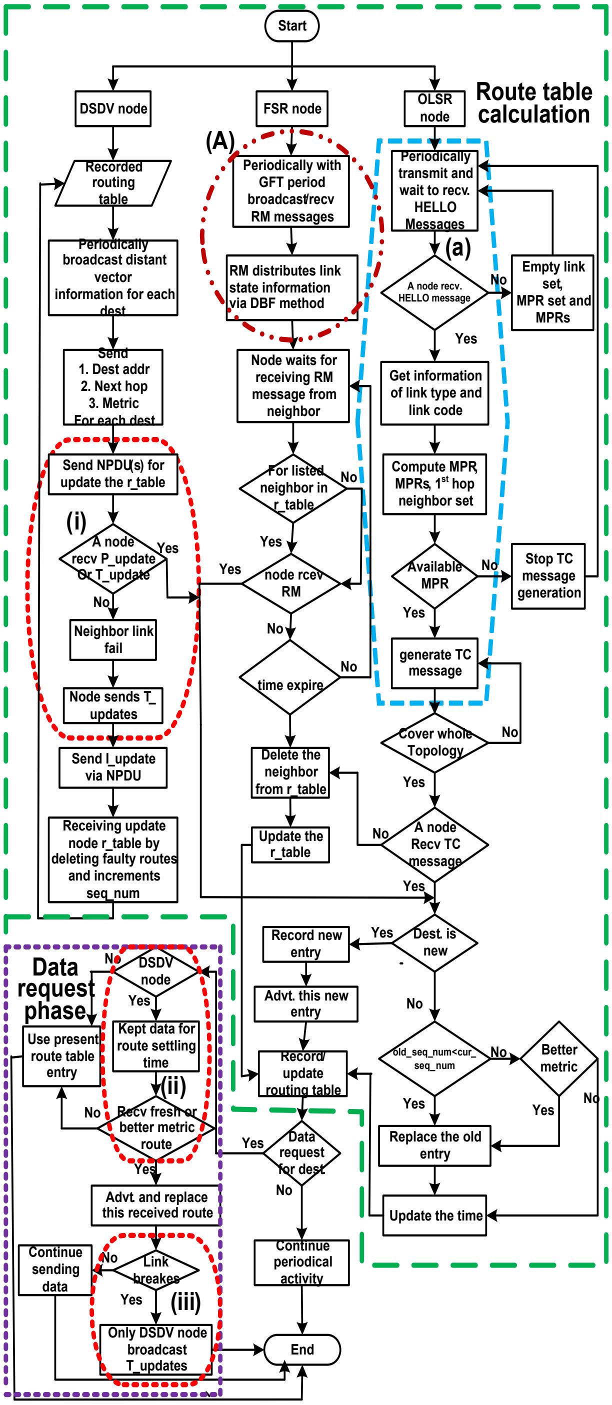


Figure 2.2: Proactive protocols, 'route calculation' and 'data request'

with the identical sequence number.

To reduce the amount of routing information carried by routing packets, DSDV uses two types of packets: first type carries all available routing information, called a *full dump* and second type carries only information changed since the last full dump, called an *incremental*. An *incremental* should fit in an *Network Protocol Data Units (NPDUs)*, as illustrated in step(i) of Fig.2.2. Moreover, *(NPDUs)* are used to control the network overhead, by arranging the "incremental" and "full dumps" utilizing the bandwidth.

Fish-eye state (FS) technique in FSR leads to the major reductions, like routing overhead caused by exchanging the whole *routing table* for convergence. As the network size grows, a *GF* update plan is used across the multiple scopes to keep the overhead low. Moreover, it updates link state information with different frequencies depending on the scope distance. By retaining a route entry for each destination, FSR avoids the additional labor of 'finding' the destination as in on-demand routing thus it maintains low single packet transmission latency. Whereas, *MPR* mechanism is used along with link state algorithm to achieve the optimization for purpose of flooding the messages and to reduce number of retransmission while forwarding broadcast packets. For link monitoring, OLSR periodically exchanges the 'HELLO messages'.

The next section describes the simulation model and simulation parameters taken into consideration to carry out this evaluation study.

2.3 Simulation Model

For the simulation setup, we have chosen Continuous Bit Rate (CBR) traffic sources with a packet size of 512 bytes. The 20 source-destination pairs are spread randomly in the network. The mobility model used is Random Waypoint. The area specified is 1000m x 1000m field presenting a square space to allow the 50 mobile nodes to move inside. A square area does not "discriminate" one direction of motion like a rectangular area does. On the other hand, it limits the number of hops. (4 to 6 for a default transmission range of 250m). All of the nodes are provided with wireless links of a bandwidth of 2Mbps to transmit on. Each packet in the communication during the simulation starts its journey from a random location and moves towards a random destination with the chosen speed of 2m/s in HWM, 8m/s in HRM, 15m/s in SCM and 30m/s in FCM, as discussed in section I. Once the destination is reached, another random destination is targeted after a specified

Table 2.1: Routing Protocols in brief

Protocol	**Distnguishd features**	**Path calculation**	**Packet forwarding**	**Flooding cntrl mechanism**	**Overhead reduction**
AODV	Local link repair	Flooding-based route discovery	Hop-by-hop Routing	Ring search algorithm	Exp. back-off alg. and grat. RREPs
DSDV	T-updates along with P-updates	DBF algorithm	Hop-by-hop routing	Exchng toplgy info. with nghbrs only	Incremental updates
DSR	Pckt salvaging of route cache	Flooding-based route discovery	Source Routing	Ring search algorithm	Exp.back-off alg.and pckt salvagng
DYMO	No use of grat. RREPs	Flooding-based route discovery	Source routing	Ring search algorithm	Exp. back-off algorithm
FSR	Multi-scope routing	DBF algorithm	Hop-by-hop routing	Grdd frquncy strategy	Fish-eye technique
OLSR	MPRs	Dijkstra's algorithm	Hop-by-hop routing	Broadcast only through selected MPRs	MPRs

pause time (from 0s to 900s). Simulations are run for 900 seconds each. Similar simulation parameters in the same mobility scenarios are used for all of the six protocols under the analysis to gather fair results. A particular scenario for a particular pause time is run for five times and mean of the five obtained values for a particular performance parameter is used to plot the graphs.

For establishing the connections among the source-destination pairs randomly, we used 'cbrgen' (can be found in the directory /nsallinone-nn/ns-nn/indep-utils/cmu-scen-gen/), which takes the type of traffic either CBR or TCP, number of nodes (50 in our case), seed (we choose 2), maximum connections and rate at which the packets are to be transferred. We have taken 20/30 sources/connections throughout the analysis. We have set all of the six protocols to transmit 1 CBR packet of 512B per second. Like connections among nodes, the mobility of mobile nodes is also random in common life. So, to perform the simulations closer to the real practice, we have used 'setdest' (can be found in the directory /nsallinone-nn/ns-nn/indep-utils/cmu-scen-gen/setdest). The binary file 'setdest' takes number of nodes, pause time (in seconds), maximum node speed in m/s, simulation time in seconds and area of the topology as length and width in meters as arguments. The implementations of AODV [28], DSDV [29] and DSR [30] used are the default ones that come with NS-2. The AODV code used is developed by the CMU/MONARCH group and is optimized and tuned by Samir Das and Mahesh Marina, University of Cincinnati. The DYMO patch used is DYMOUM from MASIMUM [31]. FSR implementation used is by Sven Jaap [32]. OLSR patch used is also UM-OLSR by MASIMUM [31], which makes it easy to configure various related parameters of the protocol, for example it runs OLSR with hop-count, ETX (Expected Transmission Count), ML (Minimum Loss) and MD (Minimum Delay) The simulations with AODV, DSDV, DSDV and OLSR protocols are run on NS-2.34 (latest NS-2 release at the time of simulations) Patches for DYMOUM and FSR are available for NS-2.29 and NS-2.30, respectively. The versions of NS-2 used for simulations include several MAC and PHY layer enhancements and performance improvements [33]. All packets (both data and routing) sent by the routing layer are queued at the interface queue until the MAC layer can transmit them. The interface queue is maintained as a priority queue with two priorities each served in FIFO order. Routing packets have higher priority than data packets. All the simulations are run on a Dual Core system with 1.86 GHz processors with 1 GB of RAM over the operating system Fedora9. In the coming four sections the behavior of the six routing protocols is analyzed with the simulation parameters taken into account for this study.

In the next four sections, the behavior of six routing protocols is analyzed with the simulation parameters taken into account for this study.

2.4 Throughput

It is amount of data successfully transferred from source to destination.

$$Throughput_{avg} = \frac{\sum_n ReceivedPackets}{\sum_t Time} B/s \tag{2.1}$$

To well estimate this metric, we only consider the received packets in our computation.

2.4.1 Throughput achieved by reactive protocols

In high mobilities, DSR possess maximum throughput except in FCM at 0,100, 200 pause times, where AODV attains more throughput. In very high dynamic situations RC of DSR becomes ineffective.

As there is no mechanism to delete the *stale routes* from RC except the RERR messages; so, the protocol fails to converge at this mobility/speed. While AODV checks the *route table (RT)* with valid time and avoids to use the invalid routes from routing table. The HELLO messages and LLR make able the protocol to handle the highest rates of mobility. The overall convergence in all other situations, DSR produces the highest throughput because it does not generate more routing packets, like AODV. RC stores multiple routes for the same destination and thus during frequent link breakage, more routes are available. Whereas, AODV's *(RT)* stores one route for one destination which is also associated with a time period. Furthermore, promiscuous listening mode provides efficient mechanism to handle dynamic situation. The worst behavior of DYMO among reactive protocols in response to mobility by showing overall less throughput value is noticed in Fig.2.3. The absence of *grat. RREPs* and dissemination of source route information collectively result in low throughputs as compared to rest of two protocols.

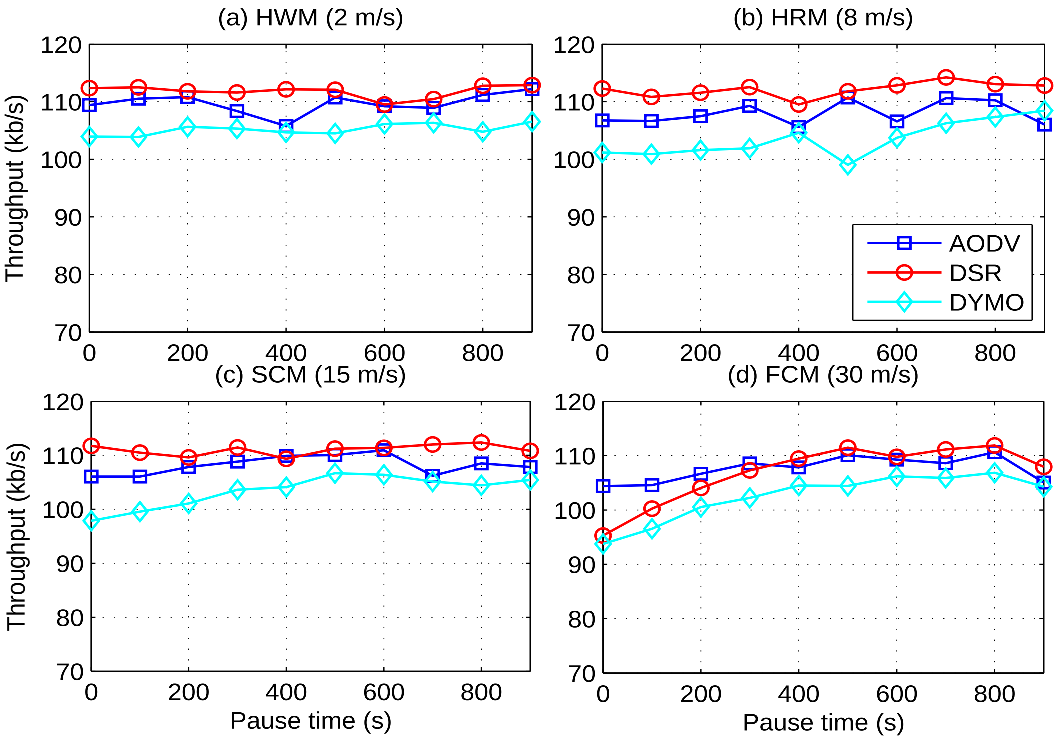


Figure 2.3: Throughput achieved by reactive protocols for varying speeds and mobilities

2.4.2 Throughput achieved by proactive protocols

Among proactive protocols, DSDV attains the highest throughput and shows efficient behavior in all mobility scenarios. The reasons for this good throughput include: firstly, when the first data packet arrives, it is kept until the best route is found for a particular destination. Secondly, a decision may delay to advertise the routes which are about to change soon, thus damping fluctuations of the route tables. The re-broadcasts of the routes with the same sequence number are minimized by delaying the advertisement of unstabilized routes. This enhances the accuracy of valid routes resulting in the increased throughput of DSDV in all types of mobility rates, as depicted in Fig.2.4.

Whereas, due to low convergence of OLSR in high mobility, there is a gradual decrease in overall throughput because increasing mobility increases the unavailability of valid routes due to its proactive nature. In static situation, in all of the four models, throughput is better as compared to moderate and relatively high mobility due to availability of stable entries for *MPRs*. Moreover, FSR and OLSR do not trigger any control messages unlike DSDV, when links break.

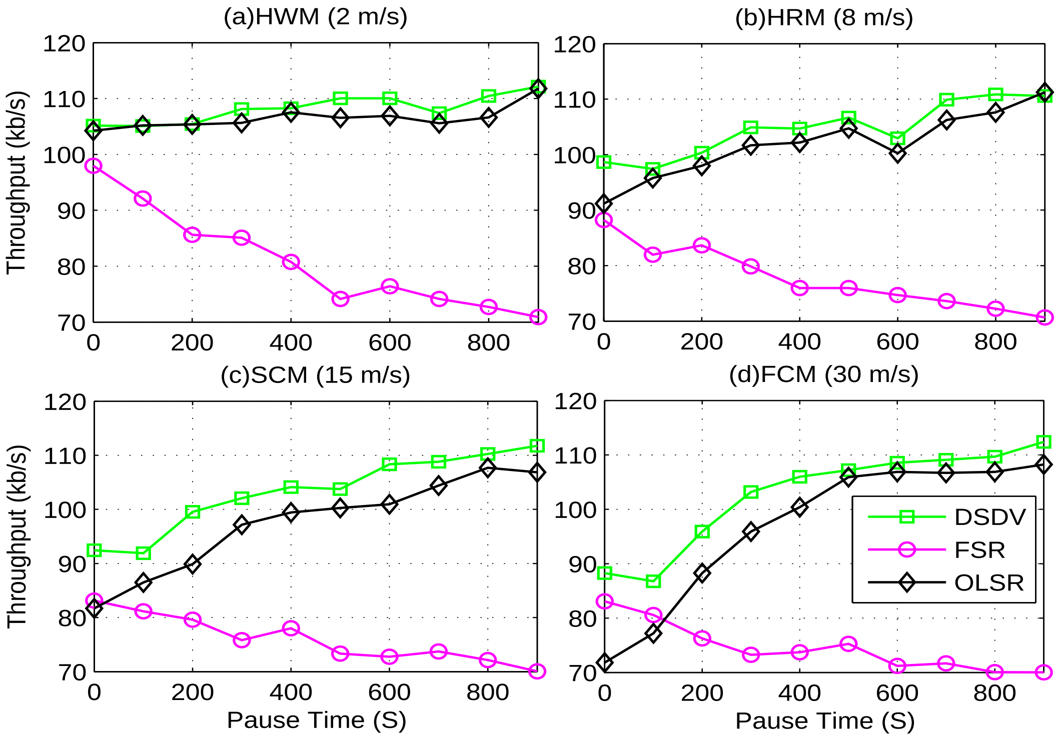


Figure 2.4: Throughput achieved by proactive protocols for four mobility models

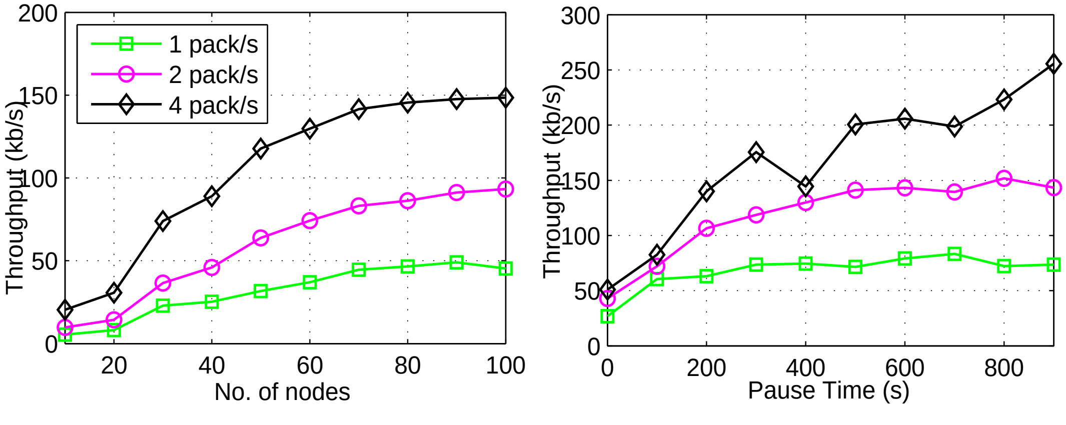
2.4.3 Interesting facts regarding throughput

Reactive protocols attain more throughput than proactive ones in high rates of mobility and speed. Reason is obvious, as proactive protocols perform route calculation before data transmission unlike the reactive ones. So, in this case if a data packet is on a calculated route and due to mobility, a link breaks, the respective proactive protocol has to perform route calculation from scratch as shown in Fig.2.2 that *RT calculation phase* take place first and then response to *data request phase* is given, which degrades the performance. All of the six protocols achieve the throughput in the order as follows: $DSR > AODV > DSDV > DYMO > OLSR > FSR$.

DSR with the highest speeds/mobilities achieve less throughput values for the reasons that at step(a) in Fig.2.1, *RC* is checked each time for a route request. Secondly, step(a) also depicts that *RC* is not associated with any explicit mechanism to delete the stale routes except the response of RERR messages. While in AODV, fresh routes are only considered, as demonstrated in step(i) of Fig.2.1. So, AODV better converges in this situation than DSR. Moreover, mobility breaks the links which generates a storm of RERR messages consuming the bandwidth because of source route dissemination and causes more drop rates.

DSDV sends more number of data packets than rest of the protocols with the lowest speed of 2m/s, at 0s pause time. Because, routes with the same sequence number are not retransmitted until the route becomes stabilized, as shown in Fig.2.2, step(ii) in the *data request phase*.

DSDV's throughput decreases at high mobility when speed increases. As, simultaneously increasing speed and mobility increases inconsistency in *RT* calculation which leads to decrease in throughput as obvious from b, c, d, in Fig.2.4. DSDV achieves the same throughput values at all speeds and at moderate and no mobilities because in less mobility size of an *incremental* becomes equal to size of a *NPDU* to make the next *incremental* smaller. For example, when a stabilized route shows a new sequence number for the same destination but the metric remains the same then this change is supposed to be non-significant and is decided to be advertised after stabilization.



(a) Scalability analysis

(b) Traffic analysis

Figure 2.5: FSR performance analysis with varying packet rates and scalabilities

FSR's strange behavior: *(i) Though it is proactive but its throughput is decreasing with* *decreasing mobility* because in low mobilities multiple routes are available in *RC*. There is lack of any mechanism to delete expired stale routes in FSR, like DSR, or to determine the freshness of routes when multiple routes are available in route cache, like AODV. *(ii) It is showing least throughput among all protocols*. The reasons include: firstly, for higher traffic rates (large number of packets per second) FSR works well [24]. It has been depicted in Fig.2.5.a. by simulating a scenario with 50 nodes moving at 20m/s speed. It is obvious from Fig. that FSR with large number of packets achieves more throughputs. Secondly, FSR is best suited for large scale multi-hop wireless networks, as the *scope update scheme* can benefit in reducing the number of routing update packets and achieve high data packet to routing packet ratio. This fact is demonstrated in Fig.2.5.b. where we have simulated FSR with 20 m/s node speed for varying number of nodes, 10, 20, . . . , 100.

DSR achieves maximum average $throughput_{avg}$ among all six protocols, as, during higher mobility, less RERR messages and RREQ messages are to be sent due to availability of valid routes in *RC*. The promiscuous mode of DSR, as described in Fig.2.1, step(b), makes able this protocol to handle the high mobility.

2.5 End-to-end Delay (E2ED)

It is the time a packet takes to reach the destination from the source. We have measured it as the mean of Round Trip Time taken by all packets.

$$E2ED_{avg} = \frac{\sum_{n=1}^{N} RTT_n}{N} \ s \tag{2.2}$$

Where N is the total number of successfully received packets.

2.5.1 E2ED produced by reactive protocols

As demonstrated in Fig.2.6, AODV among reactive protocols attains the highest delay. Because *LLR* for link breaks in routes sometimes result in increased path lengths. DYMO produce the lowest $E2ED_{avg}$ among reactive protocols because it only uses the *ERS* for route finding that results less delay; as checking the *RC* in (DSR) and *RT* in (AODV) before route discovery through *ERS* attains a some delay. At higher speeds, DSR suffers

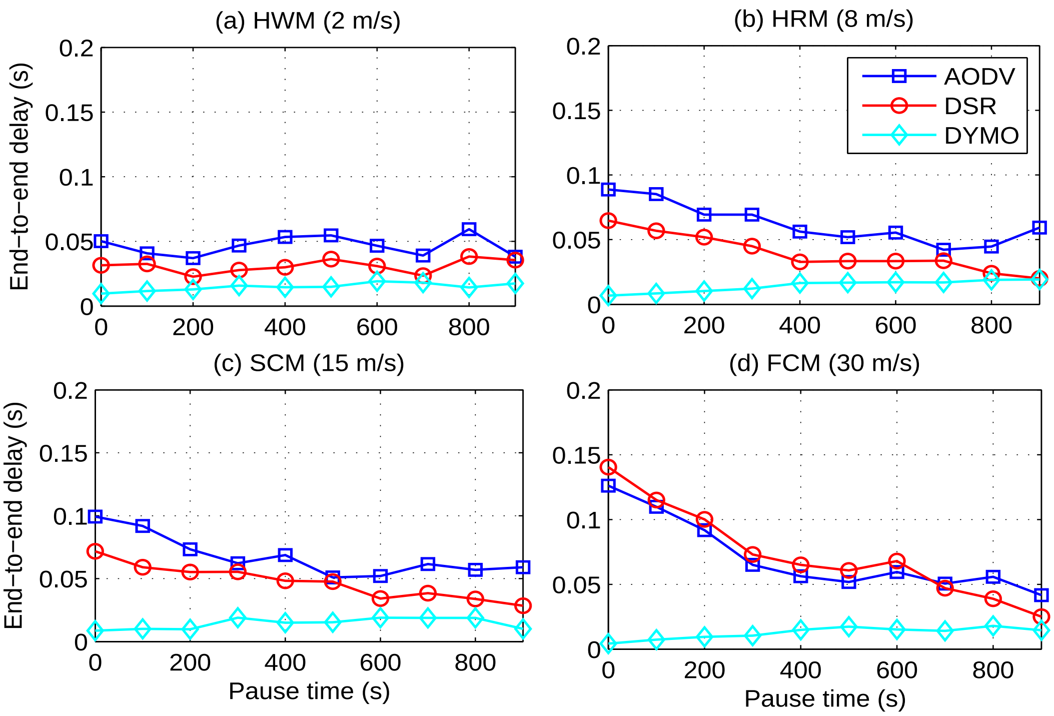


Figure 2.6: End-to-end delay caused by reactive protocols

the higher AE2ED. The reasons include: for *RD*, it first searches the desired route in *RC* and then starts *RD*, if the search fails. As, DSR does not implement *LLR*, so its AE2ED is less than AODV but during moderate and high mobility at high speed *RC* search fails frequently and results in increased delay.

2.5.2 E2ED produced by proactive protocols

In all proactive protocols, E2ED value is directly proportional to speed and mobility, as depicted in Fig.2.7. The proactive protocols have more AE2ED as compared to the reactive ones, as they calculate *RT* before data transmission. DSDV possess the highest E2ED among proactive protocols in moderate and no mobility situations, as well as in all cases its E2ED is higher than OLSR. Because DSDV keeps a data packet until it receives a good route creating delay. Furthermore, advertisements of the routes which are not stabilized yet, is delayed in order to reduce the number of rebroadcasts of possible route entries that normally arrive with the same sequence number.

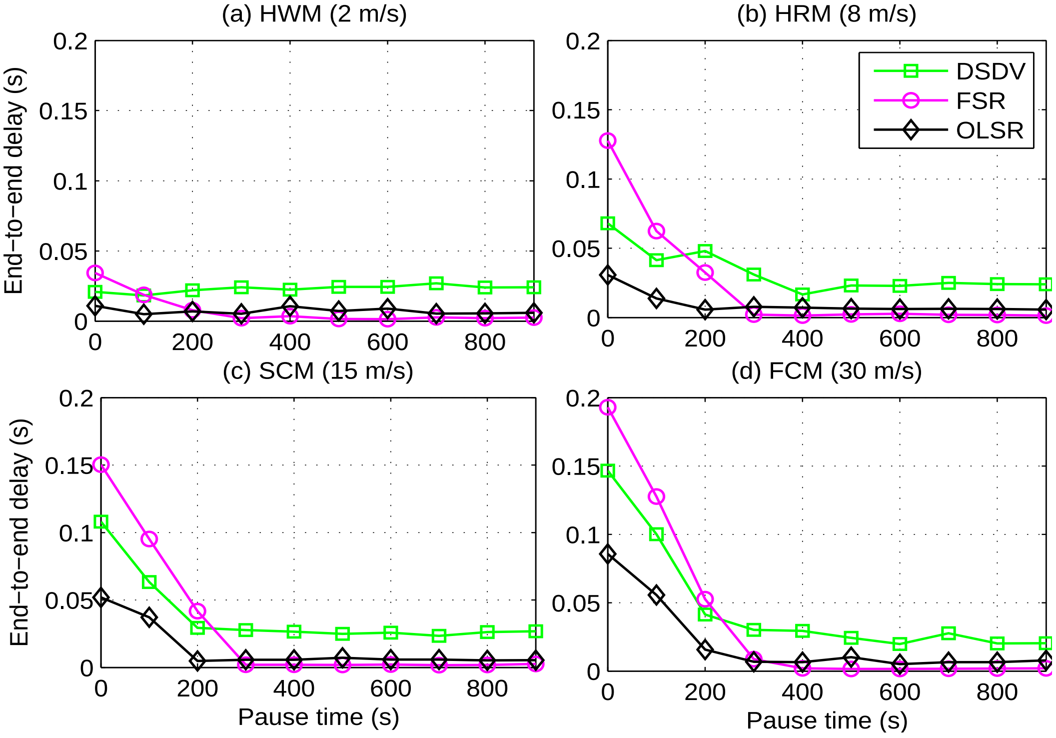


Figure 2.7: End-to-end delay by proactive protocols

FSR at higher mobilities, possess the highest AE2ED among proactive protocols. Due to *GF* mechanism when mobility increases, routes to remote destinations become less accurate. However, when a packet approaches its destination, it finds increasingly accurate routing instructions as it enters sectors with a higher refresh rate. At moderate and no mobilities at all speeds, value of end-to-end delay is the same as well as this delay is less than other proactive protocol. It is due to retaining a route entry for each destination, that avoids extra work of ”finding” the destination as in on-demand routing thus maintains low single packet transmission latency.

2.5.3 Interesting facts regarding E2ED

Generally, reactive protocols cause more delay as compared to the proactive ones. E2ED generated by all 6 protocols is: $AODV > DSR > FSR > DSDV > DYMO > OLSR$, which means that when talking about E2ED, OLSR out performs rest of the five protocols.

E2ED of DYMO is less not only among the reactive protocols but also from DSDV and FSR because it neither adapts strategy of *RC* like DSR (step(a), Fig.2.1) nor *LLR* mechanism like AODV (step(iii), Fig.2.1). Moreover, DYMO uses *ERS* algorithm which is more efficient for reducing E2ED as compared to *GF* of FSR and waiting for the best route mechanism in DSDV.

0s to 300s pause times, FSR's delay increases, as *GF* algorithm which helps to achieve higher throughputs and lower E2ED and routing overheads for large number of nodes (as, 300 and 500 source-destination pairs [24]). But, we are running FSR for only 50 nodes (20 source-destination pairs), so, *GF* algorithm produces large latencies from the stations afar. Secondly, FSR does not trigger any control messages to react for the link breaks, it uses only periodic advertisement for newly available routes.

DSR has the highest E2ED in FCM at moderate and high speed. Because at high speed, for unreachable destinations, *ERS* algorithm (step(B) of Fig.2.1.) produces delay to calculate valid routes. As DSR works well in moderate and relatively high rates of mobility, it has to compromise on delay to calculate valid routes.

AODV suffers from maximum $E2ED_{avg}$. As, *LLR* mechanism is initiated after link breakage detection. In *RM* phase, step(iii) of Fig.2.1 is demonstrating that starting of *LLR*, sometimes results in increased path lengths.

OLSR achieves the lowest E2ED. When comparing with proactive protocols, OLSR generates periodic HELLO and Topology Control (TC) messages to check links as well to compute the *MPRs* (*RT calculation phase* step(a) in Fig.2.2.) to better reduce the delay as compared to periodic exchange of whole table with the neighbors in FSR and periodic and trigger updates in DSDV.

2.6 Normalized Routing Load (NRL)

NRL is the number of routing packets transmitted by a routing protocol for a single data packet to be delivered successfully at the destination.

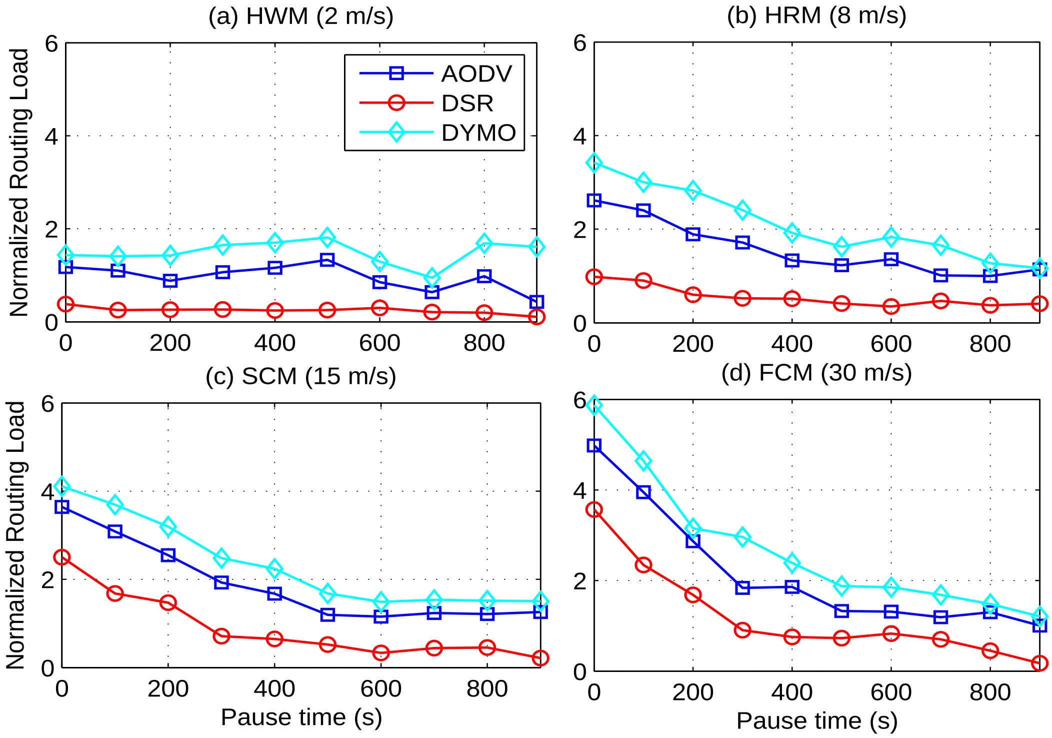


Figure 2.8: Routing overhead generated by reactive protocols

2.6.1 NRL generated by reactive protocols

Due to the absence of gratuitous RREPs, DYMO produces higher routing overhead than not only reactive protocols but also DSDV and FSR. Whereas, DSR, due to the promiscuous listening mode has the lowest routing load. Although, AODV uses gratuitous RREPs but due to the use of *HELLO messages* like DYMO and *local link repair*, it causes more routing load than DSR. One common noticeable behavior of all reactive protocols is that at high speeds and/or high mobilities, routing overhead is higher as compared to moderate and low mobilities and/or speeds. Because, in response to link breakage, all of the on-demand protocols disseminate RERR message to inform the route request generator about the faulty links and prevent the use of invalid routes. As in high dynamic situations, the link breakage is frequent, so, more RERR messages are generated resulting in high NRL.

2.6.2 Routing overhead produced by proactive protocols

Fig 2.7 shows that OLSR due to computation of MPRs through TC and HELLO messages results in the highes generation rate of routing packets. The lowest NRL is produced by DSDV, because, *incremental* and *periodic updates* through *NPDUs* reduce the routing overhead. Moreover, FSR has lower routing overhead than OLSR because it prefers *periodic updates* instead of *event driven exchanges* of the topology map which greatly helps in reducing the control message overhead during high mobility rates. Also, in FSR link state packets are not flooded. Instead, nodes maintain a link state table based on the up-to-date information received from neighbor nodes and are periodically exchange it with their local neighbors only (no flooding).

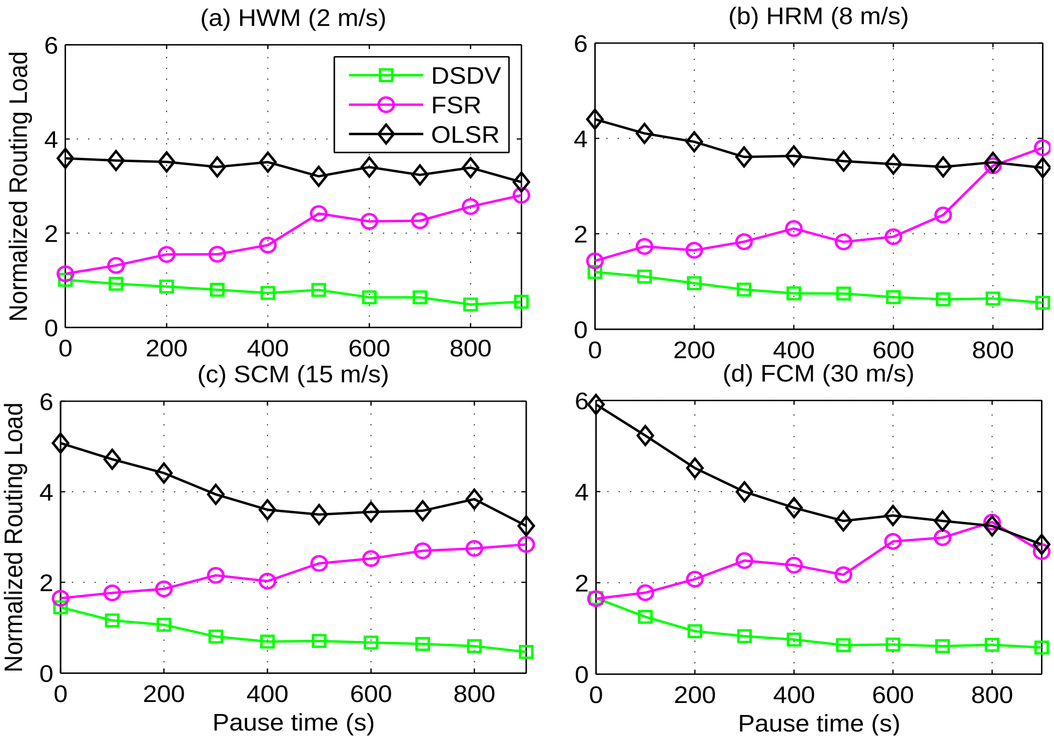


Figure 2.9: Routing overhead by proactive protocols

2.6.3 Interesting facts regarding routing load

Generally, both classes of protocols; reactive and proactive have to suffer from routing load during higher mobilities and at higher speeds. Following order depicts routing over head of six protocols in which OLSR suffers from the highest number of routing packets: $OLSR > DYMO > FSR > AODV > DSDV > DSR$.

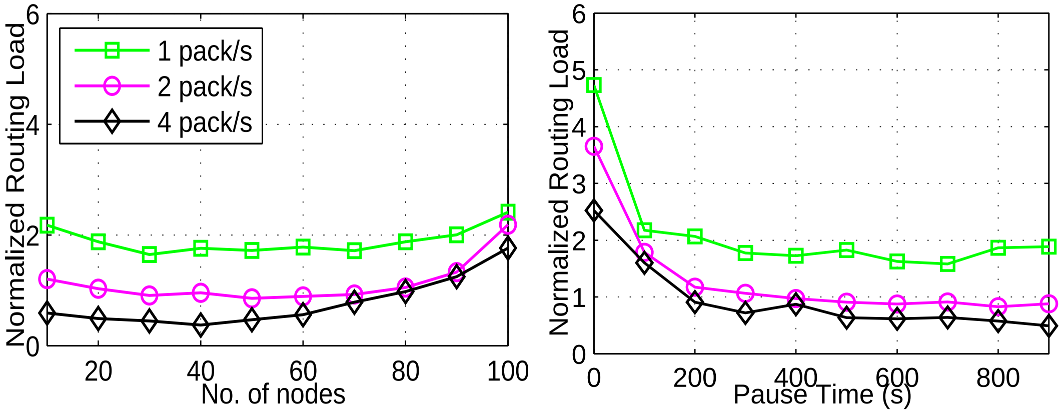
AODV possesses more NRL than DSR during all cases of mobility, because an AODV node offers connectivity information by broadcasting local HELLO messages unlike DSR.

DYMO gives higher NRL value among all reactive protocols for the reason that though *ERS* algorithm is used to reduce the routing overhead, but AODV and DSR generate *grat. RREP* messages. As demonstrated in step(iii) and in step(c) in *RM phase* of Fig.2.1, which possibly avoid the second *RD*. These messages are not generated in DYMO causing higher generation rates of routing packets than both AODV and DSR. On the other hand, *RC strategy* further reduces NRL of DSR as compared to AODV.

FSR's routing overhead is increasing with decrease in mobility. The reason is that availability of routes in *RC* is inversely proportional to mobility, i.e., in low mobilities more routes are available. There is lack of any mechanism to delete the expired stale routes in FSR or to determine the freshness of routes when multiple routes are available in route cache. These multiple routes not only increase the NRL but also affect throughput. The reasons for strange throughput of FSR are equally valid for routing load. With the same simulation scenarios as carried to justify FSR's strange throughput in Fig.2.5, we justify the strange NRL of FSR, as shown in Fig.2.10.

OLSR suffers maximum ANRL. In general, a minimal *MPR* set produces least routing overhead. But there are situations in which overhead can be a traded-off. For example, a node may decide to increase its *MPR* coverage if it observes many changes concerning the information of its neighbors caused by the mobility. However, in certain OLSR options, some control messages may intentionally be sent in advance (TC or HELLO messages, as, shown in step(a) of fig.2.2) to increase the robustness of the protocol against the topological changes.

DSR achieves the smallest NRL among all protocols. Because, along with *ERS* technique, DSR uses *RC* strategy. This allows multiple routes (step(a), *RD phase*, Fig.2.1) for the same destination to be cached which avoids overhead to perform a new *RD* each time a route in use breaks. Secondly, sender of a packet first selects and then controls the path



(a) Scalability analysis

(b) Mobility analysis

Figure 2.10: FSR performance analysis with different packet rates

used for its own packets, which, together with support for multiple routes, supports load balancing. If more route are available, then routing overhead decreases.

2.7 Performance Trade-offs made by protocols

In this section, referring to routing techniques,upon which the routing protocols are implemented, we discuss the performance of the routing protocols they achieve and price they pay. Trade-offs, the routing protocols have to make, are listed in the table.2.2.

AODV reduces PLF (or increases throughput) at the cost of routing load (and delay). Connectivity does not allow packets to drop, so, to maintain the connectivity, AODV nodes broadcast local HELLO messages every 1000 milliseconds to continuously check connectivity of active routes. *LLR* starts in case of link breakage (step(iii), Fig.2.1). This action reduces the chances of packet drop but increase the routing overhead and path lengths.

DSDV achieves throughput at the cost of delay. To ensure the best paths, nodes running DSDV always prefer new entry (for a new destination) before making subsequent forwarding decisions, i.e., data packet are kept for the best path to arrive. The *route settling time* is used to decide, how long to wait before such advertisements using average settling time, as described in step(ii) of Fig.2.2 during *data request phase.* The strategy becomes pretty beneficial when a possibly unstable route is advertised immediately after its reception.

Table 2.2: Performance trad-offs made by routing protocols

Protocol	**Modification to routing technique**	**Advances achieved**	**Price to pay**
AODV	Sequence number along with local link repair	Hi thrupt (Fig.2.3.d, 0s) in hiest mobility/hiest speed	causes delay due to local link repair (Fig.2.6)
DSDV	Sequence no. with avg. settling time	hiest thrupt when mobility is hi and speed is lo (Fig.2.4.a,0s)	causes delay due to avg. settling time (Fig.2.7,0s)
DSR	Route cache technique	Caches learned routes and increase throughput. (Fig.2.3)	Causes delay when link breaks are frequent. (Fig.2.6.d, <600s)
DYMO	Without route cache and gratuitous route reply	Reduces E2ED in hi mobility and in high speed. (Fig.2.3)	Decreases throughput. (Fig.2.6.c.d) and NRL when speed and mobility is high. (Fig.2.8.c.d. < 500s)
FSR	Multipath routing, Fisheye scopes with graded frequency mechanism	More thruput in hi mobility as comprd to lo moblty.(Fig.2.4, <400) and decrease in NRL. (Fig.2.4. < 400s)	Less throughput and increased E2ED during hi mobility and speed. Fig.2.8. > 600s and Fig.2.7.b.c.d <300s
OLSR	MPR calculation	Lo E2ED with more thruput (Fig.2.4, >300) in medium or no mobility or when speed is lo (Fig.2.9, >300s)	Highest NRL, due to MPR's computation. (Fig.2.7.)

DSDV also achieves convergence at the cost of routing load, as it generates and receives trigger as well as periodic updates demonstrated in Fig.2.2 under step(i) and step(iii) (T-update and P-update). Due to frequent link breakage during highly mobility, DSDV generates trigger updates (as in reactive protocols) along with periodic updates, that is why routing load increases to reduce Packet Loss fraction (PLF).

DSR achieves throughput at the cost of delay. DSR first searches *RC* for already learned routes for a request. If no route is found then it generates RREQ messages for desired destination. As *RC* stores multiple routes for one destination, there are more chances for presence of routes. Step(a) in Fig.2.1, results delay but achieves higher average throughput due to reduction in the chances of RREQ generation, if the route cache contains a valid route for the desired destination.

DYMO reduces E2ED at the cost of increased drop rate (or decreased throughput). The *RD phase* starts with *ring search route discovery* and is accompanied with *EB algorithm*, i.e., step(A) and step(B) of Fig.2.1. DYMO does not use *LLR* like AODV. So, it reduces E2ED but its drop rate increases when link breakage occurs frequently.

DYMO reduces routing overhead at the cost of throughput. Unlike DSR and AODV, DYMO does not generate *grat. RREPs*. This reduces routing load but decreases throughput.

FSR reduces NRL but decreases throughput. Instead of event driven updates (as in DSDV, step(iii), Fig.2.2), FSR uses $P_updates$ to exchange topology map by greatly reducing the control message overhead but causing more packet drop (decreasing throughput).

OLSR reduces $E2ED_{avg}$ *and achieves throughput at the cost of routing overhead.* In moderate and no mobility, minimal *MPR* computation generates more routing packets but the connectivity provided by this way not only guarantees the throughput but also helps OLSR to achieve the least E2ED among all of the six protocols.

OLSR also achieves convergence at the cost of routing load. In certain options (TC or HELLO messages), some control messages are intentionally sent in advance to increase robustness of the protocol against topological changes. This causes local increase of control traffic.

In Table.3.3, based upon the simulation results in NS-2, all of the six protocols are ranked.

Table 2.3: Ranking of the protocols based upon the simulation results

Mobility	**Throughput**	**E2ED**	**NRL**	**PLF**
No/Low mobilities (700s-900s pause times)	DSDV	AODV	OLSR	FSR
	AODV	DSR	FSR	DYMO
	DSR	DSDV	DYMO	OLSR
	OLSR	DYMO	AODV	DSR
	DYMO	OLSR	DSDV	AODV
	FSR	OLSR	DSR	DSDV
Moderate mobilities (300s-600s pause times)	DSR	DSR	OLSR	FSR
	AODV	AODV	FSR	OLSR
	DSDV	DSDV	DYMO	DYMO
	DYMO	DYMO	AODV	DSDV
	OLSR	OLSR	DSDV	AODV
	FSR	FSR	DSR	DSR
High mobilities (0s-200s pause times)	DSR	FSR	OLSR	FSR
	AODV	DSDV	DYMO	OLSR
	DYMO	AODV	AODV	DSDV
	DSDV	DSR	FSR	DYMO
	OLSR	OLSR	DSDV	AODV
	FSR	DYMO	DSR	DSR
All mobilities (0s-900s pause times)	DSR	AODV	OLSR	FSR
	AODV	DSR	DYMO	OLSR
	DSDV	FSR	FSR	DSDV
	DYMO	DSDV	AODV	DYMO
	OLSR	DYMO	DSDV	AODV
	FSR	OLSR	DSR	DSR

2.8 Conclusion

The massive simulations of the chosen protocols have demonstrated that reactive protocols are superior to the proactive ones, provided that mobility is taken into account as a constraint. Nodes running AODV send data packets merely carrying addresses of the destination unlike DSR that requires data packets to carry the source routes also. So, DSR has more overhead in bytes than AODV. On the other hand, DSR has less overhead in terms of number of packets. AODV broadcasts periodic HELLO messages and sends more control messages than DSR to find and repair the routes by *LLR* technique, so, it produces more routing load than DSR. This can be concluded that AODV and DSR show the best performance during all mobilities and at all speeds. This study recommends that DSR can be selected for networks which are conscious about the number of hops, where the traffic is overhead-sensitive and nodes favor packet overhead on the low byte overhead. AODV should be chosen where the number of hops is not a problem and the nodes prefer low byte overhead on the packets. For delay sensitive applications, DYMO in reactive protocols and OLSR in proactive protocols are the plausible choices. During all this evaluation, we come to realize that the most important component of a routing protocol is routing link metric, so, in future we are interested to propose and implement a new ETX-based routing link metric with AODV and OLSR, as discussed in [34].

Chapter 3

Analyzing Scalability and Traffic Loads

Contents

3.1 Introduction

The current development, popularity and prices of wireless devices have increased their use. As a result, users with wireless mobile devices are increasing in number. Examples of more populated wireless networks include; conference networking scenarios, disaster scenarios, several military applications and deployment of sensors, etc that involve hundreds to tens of thousands of wireless devices.

It has becomes a challenge for wireless routing protocols to scale to the population along with varying traffic rates. Being responsible to operate a populated network, a routing protocol is expected to handle the following issues: link break in a wireless link and sending its notification to the originator, any repair action, etc. These issues, of course, are no problem for a network with fewer nodes and small number of flows. But in case of more number of nodes a link break notification has to traverse tens of hops to reach back to the source. Route discovery is not an issue in a small network, but for larger networks, it impacts the overall network performance.

The contribution of the study includes: analytical analysis of the scalability and traffic handling properties of six routing protocols, performance leaks in each protocols due the strategies working behind the protocols in the populated networks, analytical performance comparison of the simulation results and insights that propose tracks for the future work for scalable routing protocols.

3.2 Simulations

The simulation parameters used in this chapter are as follows:

The sources transmit Continuous Bit Rate (CBR) traffic. The nodes taking part in the simulation are randomly dispersed in an area of 1000m x 1000m . The nodes move in the simulation area following the Random Waypoint Model. Each node in network starts moving towards a randomly chosen destination. The speed with which a node moves is set to be 20 m/sec. When a node reaches the destination, it takes a pause for a specified time period. The pause time in our simulation setup is 2 sec. As soon as this pause time for of a node is expired, it starts its journey towards next randomly chosen destination. This process continues until the end of the simulation time, 900 sec in case of our simulations. The bandwidth provided to all the wireless links is 2 Mbps. To examine the behavior of protocols under different *network loads*, simulations are run for packet rates of 2, 4, 8, 16, and 32 packets/sec. The size of packet is set to 64 bytes. For scalability analysis, the packet size is 512 bytes. Number of nodes used is 50, out of which 20 act as sources. For each packet rate the simulation is run five times and the result is averaged for analysis. NS-2.34 is used for all the simulations on a Dual Core system with 2.0 Ghz processors and 4 GB of RAM running Fedora 9.0.

3.3 Throughput

Throughput values achieved by both reactive and proactive protocols are calculated in the same way as section 2.3 by eq. (2.1) and are discussed in the following subsections.

3.3.1 Successful packet deliveries attained by reactive protocols

AODV shows convergence for all data rates and all scalabilities, whereas DSR is scalable but fails to converge in medium and high traffic rates while DYMO degrades its performance in more population of nodes as well as for medium and high data loads.

[18] specifies that AODV can better handle a wireless network of tens to thousand nodes and varying rates of data loads. This protocol performs better among reactive protocols for high network flows and traffic rates. The presence of gratuitous Route Replies *grat. RREPs* and time-based routing activities, as shown in Fig.3.1, step(i) that makes able the protocol to perform well by always choosing a fresher end-to-end path. The route deletion using Route Error *RERR* messages is also conformist. It also maintains predecessor list; *RERR* packets reach all nodes using a failed link on its route to any desired destination. That is why less packet loss ratio is there in AODV, this results better throughput in high loads as well as large populations.

Another reason regarding AODV to outperform DSR and DYMO is its disseminations of the distant vector information that occupy less bandwidth which is an essential for both more number of data packets are more number of flows. Routing packet in AODV contains the next hop information to destination and not complete source information, like DSR that increases message size.

DSR attains the highest throughput in low network load as well as less density, as depicted in Fig.3.3.a. This protocol uses *Routing Cache (RC)* and maintains multiple

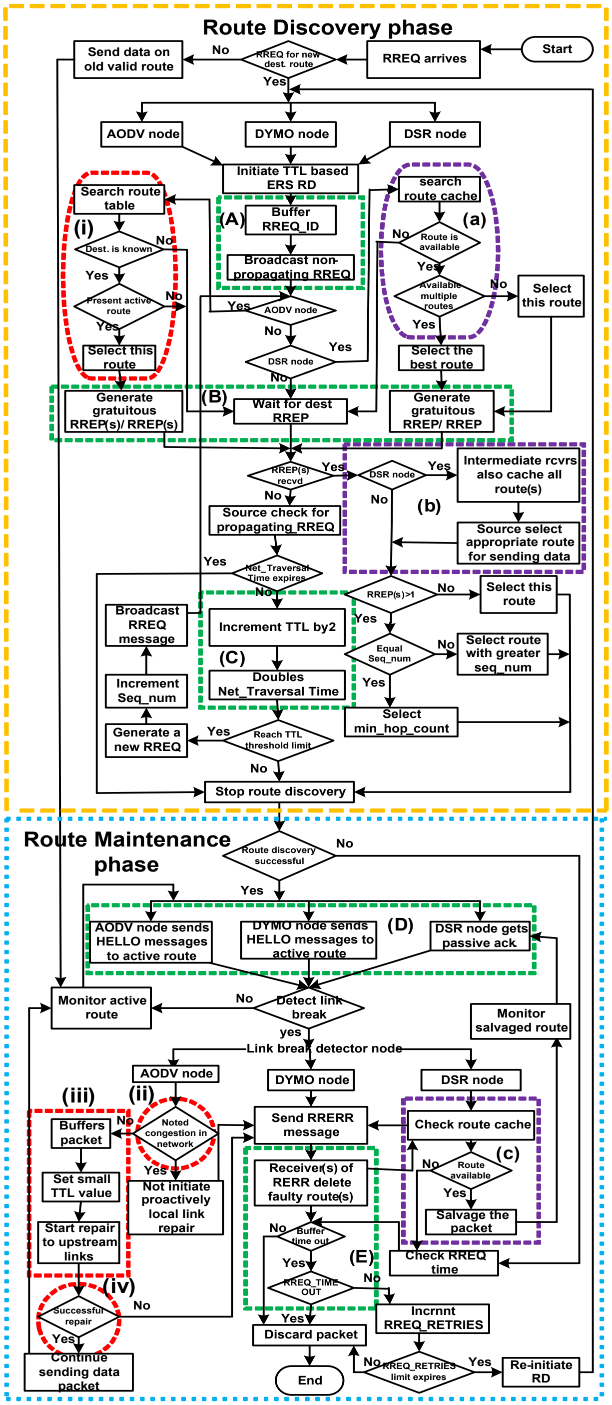


Figure 3.1: Flow chart of reactive protocols AODV, DSR, DYMO, and algorithms used

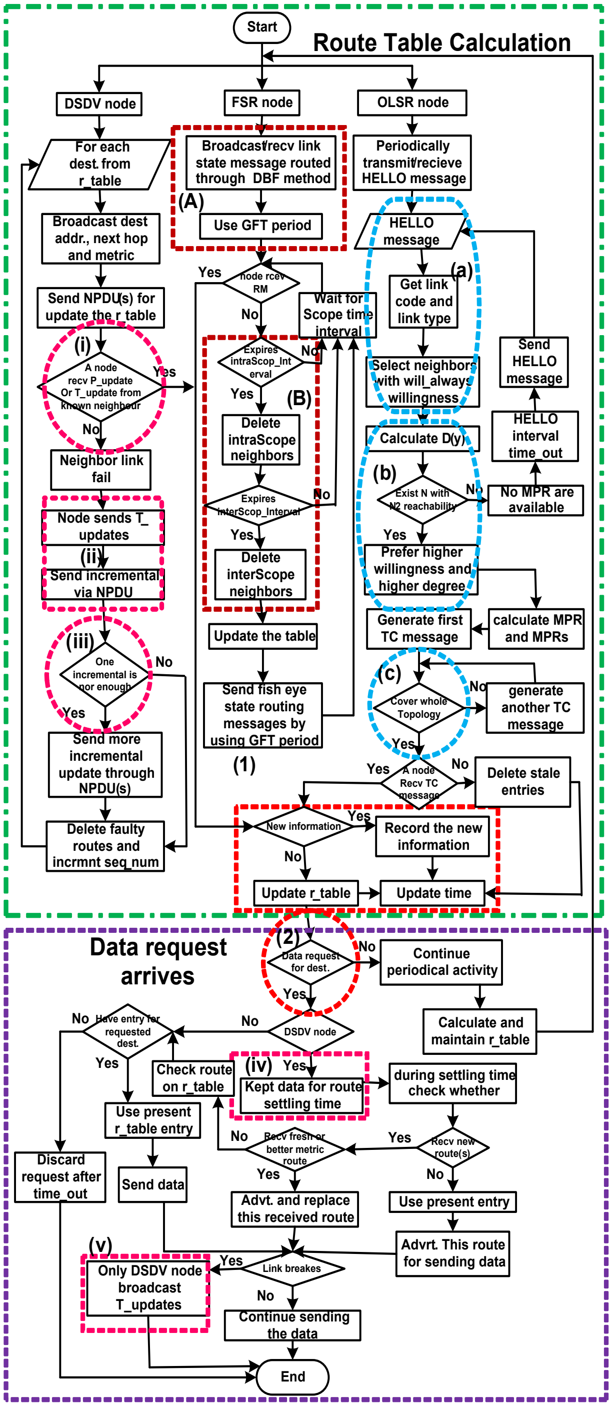


Figure 3.2: Flow chart of proactive protocols DSDV, FSR, OLSR, and algorithms used

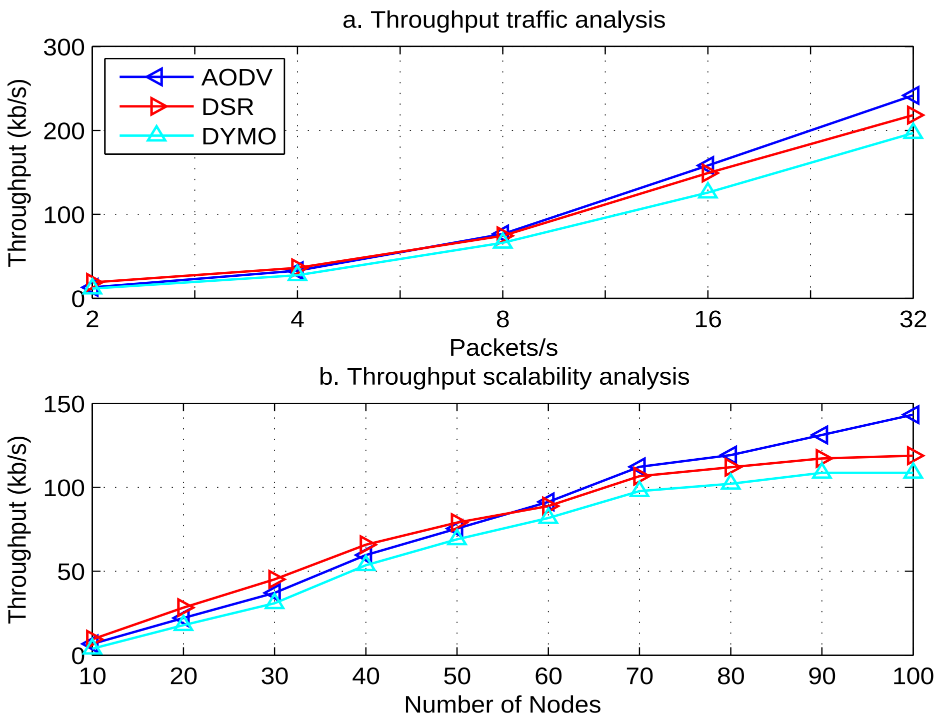


Figure 3.3: Throughput achieved by reactive protocols

routes per destination entry. Aggressive replies are insistently generated because of *RC* to all requests reaching a destination from a single request cycle. Initiator learns many alternative routes to a destination which is useful if primary route fails. DSR is also scalable as it shows convergence to all populations. All features of the protocol entirely operate on-demand allowing routing overhead to automatically scale to only what is needed to react to changes for the routes currently in use. Furthermore, it is claimed in [DSR RFC] that for DSR that it guarantees outstanding performance for up to 200 wireless nodes. Though DSR packets contain the complete source information, as compared to next hop information (distant vector routing in AODV) but Packet Salvaging *(PS)* techniques makes it more scalable as compared to DYMO. In more stress situations, i.e., in medium as well as in more data loads, DSR degrades the performance. As, it does not have any explicit mechanism to delete the expired stale routes in *RC*, except those which are deleted by *RERR messages* or prefer fresher routes. Moreover, belligerent use of *RC* consequences performance degradation when traffic load is medium or high.

The absence of *grat. RREPs* and dissemination of *source routing (SR)*packets, make DYMO less converged at medium and high traffic rates (as compared to AODV) and for high scalabilities (as compared to both AODV and DSR). Throughput line of DYMO in Fig. 3.3.a, illustrates that throughput values start decreasing when network loads increase. The reason is that DYMO does not implement any supplementary mechanism except the basic *Exponential Back-off (EB)* algorithm (used by AODV, DSR and DYMO) to handle data traffic loads, such as *grat. RREPs* in AODV and in DSR. Besides, HELLO messages to check connectivity and *SR* dissemination result more bandwidth utilization in high population and high number of flows resulting in decreased throughput. On the other hand no existence of *grat. RREPs* saves more bandwidth than DSR; due to *PS* and *grat. RREPs* broadcasts *SR* that consume network bandwidth.

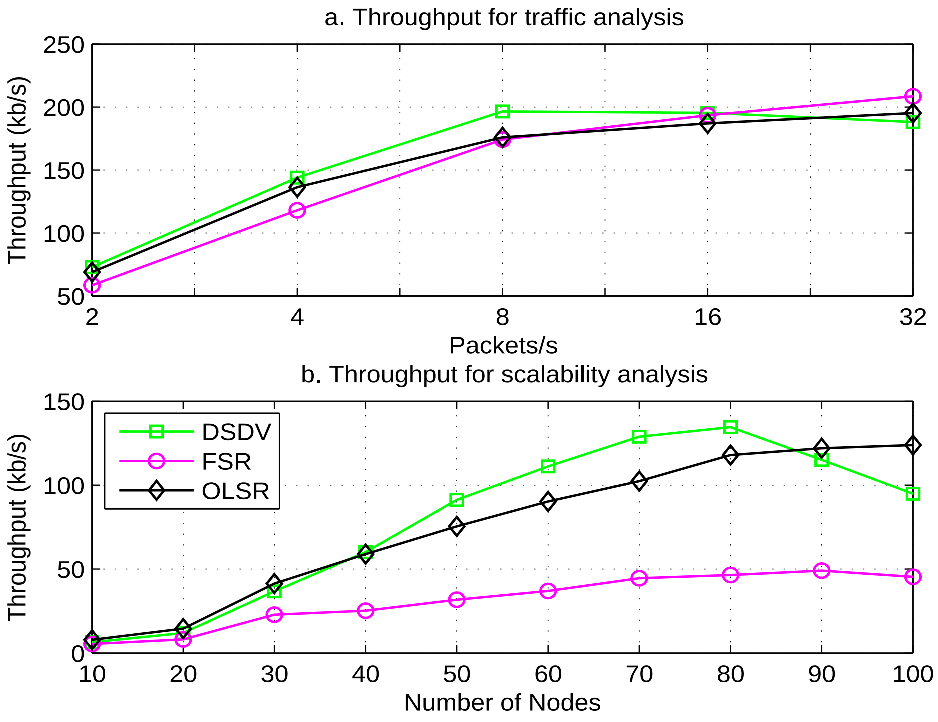


Figure 3.4: Throughput achieved by proactive protocols

3.3.2 Throughput attained by proactive protocols

FSR shows appreciable performance for varying traffic rates and OLSR is well scalable among proactive protocols. In medium and high traffic loads, FSR's performance is depicted in Fig.3.3.b. This is due to introduction of new technique of multi-level *Fish-eye Scope (FS)*, that reduces routing overhead and works better when available bandwidth is low, thus increasing throughput in case of increased data traffic loads and reduces routing update overhead. Although, DSDV uses *Network Protocol Data Units (NPDUs)* to reduce routing transparency but *Trigger Updates (T-updates)* cause routing overhead and degrade performance. OLSR uses *Multi-pont Relays (MPRs)* for reduction of overhead but computation of these *MPRs* takes more bandwidth. Therefore its throughput is less than FSR.

Moreover, through updating link state information with different frequencies depending on *FS* distance, as obvious in step(B) in Fig.3.2, FSR well scales to large sized networks. *FS* technique allows exchanging link state messages at different intervals of nodes within different *FS* distances that reduce the link state message size. Further optimization helps FSR to only broadcast topology messages to neighbors in order to reduce flood overhead. If FSR would have taken MAC layer feedback in case of link brakes then there might be exchange of messages to update neighbors, consuming bandwidth and lowering throughput. This faster discovery results in a better performance during high traffic loads.

Simulation results of OLSR in Fig.3.4.a and b show that it is scalable but less converged protocol for high traffic rates. This protocol is well suited for large and dense mobile networks, as it selects optimal routes (in terms of number of hops) using *MPRs*. Step(b) in Fig.3.2 describes the selection of the highest degree *MPR*nodes. *MPRs'* computation is used to reduce dissemination overhead which produces typical flooding process, thus occupies precious bandwidth and drops the data packets. In a dense network, more optimizations can be achieved as compared to the classic link state algorithm. *MPRs* better achieve scalability in the distribution of topology information. While in higher data flows, there is no mechanism of multi-path routing, so, this protocol cannot perform well when traffic load increases.

DSDV dilapidation is noticed in higher data loads (Fig. 3.4.a.), as increasing throughput ratio among variant loads becomes less in medium and more network loads. As, in this protocol new route entry is advertised when the subsequent forwarding data packet is requesting for the new destination. This advertisement leads to increase routing overhead and thus decreases the throughput. The protocol possess overall highest throughput values among proactive protocols, as calculated from Fig. 3.4.b, but these higher values are because of the throughput obtained in medium scalabilities. In high scalabilities, it represents lower throughputs, as route settling time increases $E2ED_{avg}$ and multiple *NPDUs* (step(iv) of Fig.3.2), increase routing load in large population. Moreover, NRL_{avg} increases due to occurrence of more *full dumps* (changing the entire routing table) that consequently affects the throughput.

3.3.3 Interesting facts regarding throughput

Both data and control packets in DSR carry SR information (complete path information from source to destination) that significantly causes overhead in larger networks. So, overhead produced by DSR packet header is larger than AODV.

AODV disseminates distant vector routing information that occupies less bandwidth unlike *SR* information. This leads better performance of AODV in higher data loads among reactive protocols.

DSR is more suitable for small populations due to the PS strategy. While in case of more number of nodes, this strategy degrades performance due to aggressive generation of routing packets.

For less dense networks

$$P_{dr} \propto Range_{dist} \tag{3.1}$$

In large topological area, when small number of nodes are dispersed, then this results in more distance in terms of range, $Range_{dist}$, as probability of being out-of-range increases. That is why more packet drop rate P_{dr} occurs.

In the case of more number of nodes in the same area:

$$P_{dr} \propto If \tag{3.2}$$

When population increases, then interference If among the contending nodes in the same transmission range is also increased.

DSDV overall attains the highest throughput among proactive protocols. As, *NPDUs* control routing overhead and delay advertisement for getting stabilized routes. But in high scalabilities; 90 stations and more, DSDV's packet drop rate increase because, delay and routing load increase by increasing the population, as a result DSDV convergence power reduces.

3.4 Average End-to-end Delay ($E2ED_{avg}$)

It has been calculated in the same way as in chapter2 in section 2.4 and by the equation 2.4.

3.4.1 $E2ED_{avg}$ produced by reactive protocols

As far as data loads are concerned, $E2ED_{avg}$ of reactive protocols is more in medium and high network loads. The *grat. RREPs* produce diverse effects in different data packet generation and different node density scenarios. DYMO does not use this strategy,therefore it suffers low delay in low traffic while produce high latency in high data rates. On the other hand, absence of the mechanism keeps the lowest E2ED of DYMO in all scalabilities. *PS* and *grat. RREPs* keep the delay low in medium and high traffic scenarios for DSR but first checking the *RC* (step(b) of Fig.3.1) instead of simple *expanding ring search (ERS)* based *RD* process augments the delay when population increases, thus more delay of DSR is presented in Fig.3.5.b. as compared to DYMO. AODV experiences the highest $E2ED_{avg}$ in all scalabilities due to *local link repair (LLR)* process; as can be seen in Fig.3.1, step(iii). Moreover, for changing traffic rates, due to the decision based on congestion for *LLR*; (step(ii) in Fig.3.1)there is decrease in latency at 16pack/s and 32pack/s as compared 8pack/s.

3.4.2 $E2ED_{avg}$ caused by proactive protocols

Increase in traffic rates and node density result more delay for all of three proactive protocols. FSR overall suffers higher delay in both situations. To retain route entries for each destination, this protocol maintains low single packet latency when traffic load or population is small. The *graded frequency (GF) mechanism* (GFT in Fig.3.2, step(A)) is used to

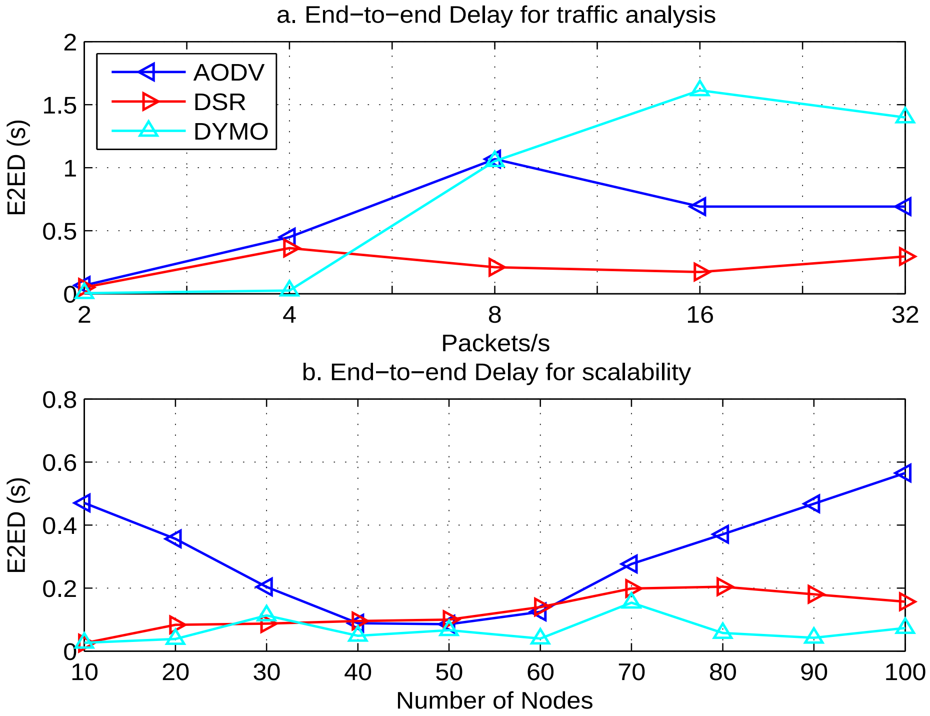


Figure 3.5: End-to-end delay produced by reactive protocols for varying traffic rates and number of nodes

find destination to keep routing overhead low. FSR exchanges updates more frequently to the near destinations. Thus, in higher data rates or more scalabilities this protocol attains more E2ED.

The reason for delay in DSDV is that it waits to transmit a data packet for an interval between arrival of first route and the best route, as depicted in step(iv) of Fig.3.2. This selection creates delay in advertising routes which are about to change soon. A node uses new entry for subsequent forwarding decisions and route settling time is used to decide how long to wait before advertising it. This strategy helps to compute accurate route but produces more delay.

A proactive protocol first calculates routing tables, so, for larger networks, it takes more time resulting in more end-to-end delay. Small values of AE2ED for OLSR are seen among proactive protocols in all scalabilities, as shown in Fig.3.6.b, because, *MPRs* provides efficient *flooding control mechanism*; instead of broadcasting, control packets are

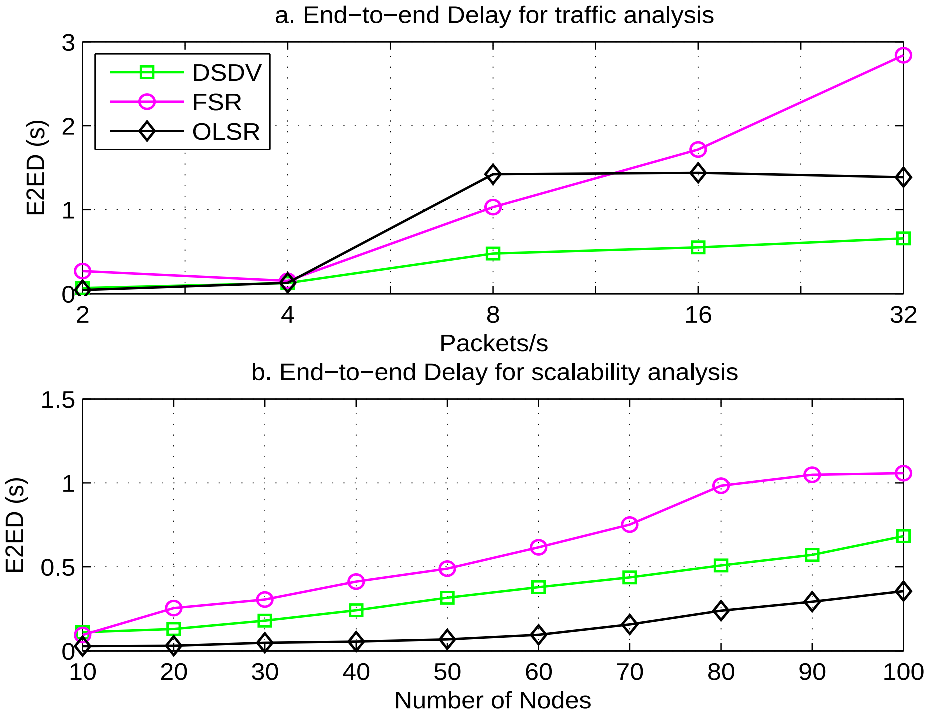


Figure 3.6: End-to-end delay prodiced by proactive protocols

exchanged with neighbors only.

3.4.3 Interesting facts regarding E2ED

In lower data traffic, DYMO possess the lowest AE2ED. Because of the absence of *grat. RREPs*, as present in AODV and DSR. Moreover, absence of *RC* also reduces the time taken for searching routes among multiple routes from *RC*, i.e., in step(a) of Fig.3.1.

Highest E2ED among reactive protocols is shown by AODV. Though, *ERS* algorithm is adopted by all reactive protocols (in the study) but this behavior of AODV is due to *LLR* strategy.

Among proactive protocols in traffic scenario, DSDV possess lowest E2ED. As, already present route entry is advertised for the subsequent forwarding messages requesting for the same destination reducing the delay.

Lowest E2ED among reactive and proactive protocols is produced by DYMO for different *node densities*. Only *ERS* algorithm is used by DYMO, while in DSR delay due to *RC* and in AODV delay due to *LLR* are introduced along with *ERS*.

3.5 Normalized Routing Load (NRL)

NRL is the number of routing packets transmitted by a routing protocol for a single data packet to be delivered successfully at destination.

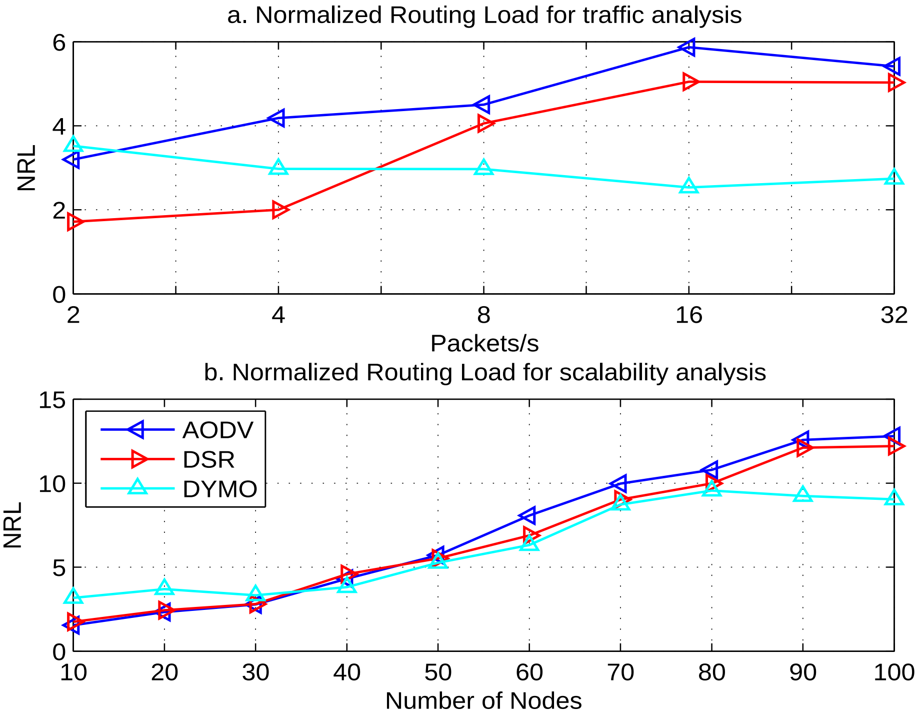


Figure 3.7: NRL generated by reactive protocols

3.5.1 Routing load generated by reactive protocols

Approximately the same comparative behavior of NRL is seen in Fig.3.7.a.b, in both traffic and population situations of all protocols. In medium and high populations and traffic rates, routing load of DYMO is less than DSR and AODV. While in medium and more density and network load, AODV attains the highest routing load. The *HELLO messages* to check the connectivity of active routes, *LLR* and *grat. RREPs* increase the generation of control packets. Whereas, *PS* (step(c) in Fig.3.1) of DSR along with *promiscuous listening mode* (step(b) in Fig.3.1) jointly reduce the routing overhead in low data load and scalability. Each node participating in *RD* process (including intermediate nodes) of DSR, learns the routes to other nodes on the route. *PS* technique is used to get routes from route cache of the intermediate nodes. This strategy is used to quickly access and to solve broken link issues by providing alternative route. However in large population of nodes, intermediate nodes generating more *grat. RREPs* increase routing overhead.

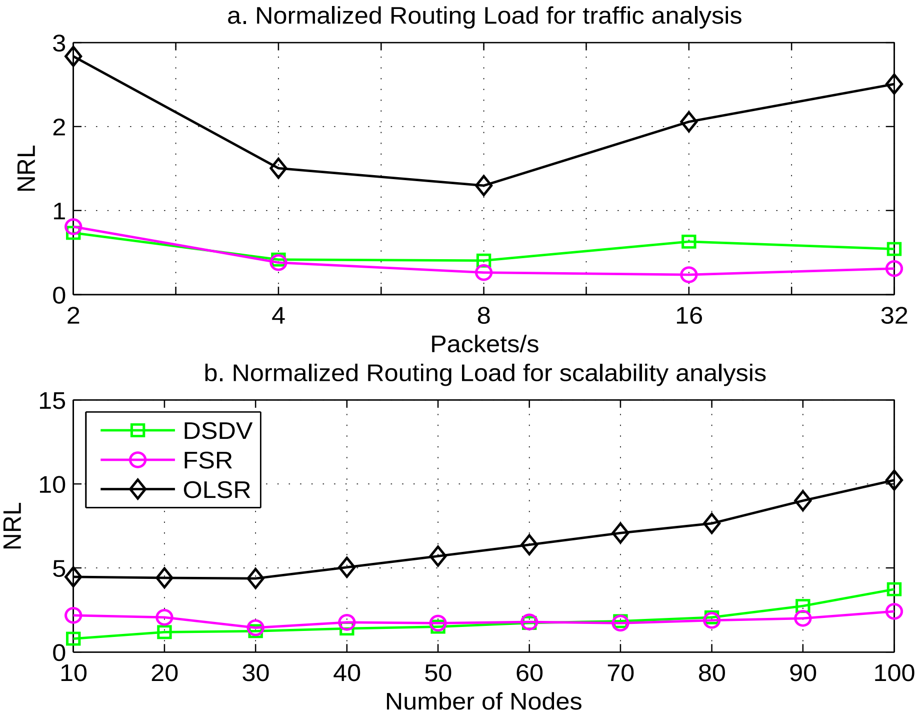


Figure 3.8: Routing packets generated by proactive protocols

3.5.2 Routing overhead generated by proactive protocols

As depicted in Fig.3.8.a.b, in all scalabilities and traffic loads, OLSR is generating the highest NRL_{avg} among proactive protocols. It happens due to *MPR* mechanism that controls the dissemination of control packets in the whole network. But calculation of these *MPRs* through *topology control (TC) messages* and *HELLO messages* (step(a) and step(c) in Fig.3.2) increase the routing load. Moreover, OLSR link state messages are used to calculate *MPRs* that generate routing overhead.

DSDV and FSR sustain low overhead in all network loads and in low and medium scalabilities. As, DSDV upholds routing table with separate route entry for new destination, while a node does not use the new entry for the same destination in making subsequent forwarding decisions. Moreover, *NPDUs* are arranged to disseminate incremental updates for maintaining low routing overhead.

Whereas, FSR reduces congestion by the help of *fish-eye scope (FS)* technique. The link state packets are not flooded, instead, nodes maintain a link state table based on the up-to-date information received from neighboring nodes. This information is periodically exchanged it with their local neighbors only (no flooding, as presented in step(A) in Fig.3.2. Using different exchange periods for different entries in routing table, routing update overhead is reduced. Furthermore, when network size grows large, a *GF* update plan is used across multiple scopes to keep the overhead low. It does not trigger any control messages when a link failure is reported (step(v) of Fig.3.2 shows behavior of proactive protocols in case of link breakage). Thus FSR is suitable for high changing topology environments.

3.5.3 Interesting facts regarding NRL_{avg}

Lowest normalized routing load among all six protocols during this analysis study is shown by DSDV and FSR. In DSDV, average settling time parameter and old stabilized valid route entry are used instead of advertising new route entry when successive messages are requesting for the same destination. When data load increases, FSR's traffic load decreases because of *FS* technique along with *GF* mechanism. Moreover, DSDV's *NPDUs* reduce routing, as their depends upon the population and topological changes.

Highest NRL_{avg} is generated by AODV among all protocols. AODV *HELLO messages*, *LLR* and *grat. RREPs* as a whole result in large number of control packets. There is no *source routing* or *promiscuous listening* is used. AODV has to rely on a route discovery flood more often, generating more networks overhead.

Gradual increase of routing overhead in DSDV: DSDV sends two types of routing updates: *periodic updates (P-updates)* and *trigger updates (T-updates)*. Previous ones carry full routing table, called, *full dump*, while later ones carry *incremental;* the information changed from last *full dump*. Meanwhile, *NPDUs* are used to carry these updates, as shown in step(3) of Fig.3.2. In small population, chance of *full dump* is reduced. So, *NPDUs* produce routing packets that gradually increase from small population to large population.

3.6 Trade-offs to achieve performance

This section describes and relates the prices paid by routing protocols to achieve efficiency for varying number of nodes and traffic rates.

At the cost of routing overhead, AODV achieves throughput for varying traffic loads and scalabilities. LLR, HELLO messages, and grat. RREPs repair links, check connectivity of active routes and receive replies from intermediate nodes, respectively, increasing the NRL_{avg}, as shown in Fig.3.7.a.b. Whereas, these strategies result in efficient performance for high scalabilities (due to successful *LLR*), (Fig.3.1, step(iv)), avoid route re-initiation and thus attains the highest throughput.

DSDV succeeds to achieve the highest throughput and lower routing overhead at the cost of increased end-to-end delay for varying packet rates. On reception of the best end-to-end route, a data packet is sent to destination. The selection of best route produces more delay, as compared to OLSR, avoiding extra routing load and packet drops.

DSR attains low delay value at the cost of routing overhead in medium and more traffic loads and in medium and high scalabilities. PS mechanism helps to avoid second *RD*. So, less delay in all offered network loads. On the other hand, DSR has no method to refresh routes, so, these stale routes are disseminated through *grat. RREPs* creating routing overhead.

DYMO does not generate grat. RREPs to keep routing load low but it lowers throughput when data traffic increases, as, it only uses *ERS* algorithm to reduce routing overhead.

Only using ERS algo., DYMO attains lowest AE2ED, but its throughput becomes less as compared to other reactive protocols in higher scalabilities.

GF mechanism in FSR, produces low NRL_{avg} *at the cost of delay in medium and high data flows and in all populations.* This frequency distribution through graded-frequency avoids flooding to the entire network. Thus reduces its routing overhead, but this graded frequency mechanism results delay.

OLSR's MPRs are only responsible for forwarding messages. producing less delay in

Table 3.1: Performance trad-offs made by routing protocols

Routing technique	**Protocol**	**Strategies**	**Traffic analysis**		**Scalability analysis**	
			Advantage achieved	**Price to pay**	**Advantage achieved**	**Price to pay**
Distance vector	**AODV**	Hello msgs, *LLR*, *grat. RREPs*	Highest throughput (due to *LLR*) Fig.3.3.a.	Produces max. NRL Fig.3.7.a.	Highest throughput (due to LLR) Fig.3.3.b	Produces max. E2ED, Fig.3.5.b.
	DSDV	Avg. settling time parameter	Low NRL, Fig.3.8.a.	Increased E2ED, Fig.3.8.a.	High throughput, Fig.3.4.b.	Increased E2ED, Fig.3.6.b.
Source routing	**DSR**	*PS* due to *RC* mechanism	Lo delay, Fig.3.5.a. Med/hi traffics	Hi NRL, 8, 16, 32 p/sec, Fig.3.7.a.	Lo delay, Fig.3.5 5.b(> 50nodes).	Increased NRL (> 50 nodes) Fig.3.7.b.
	DYMO	No *grat. RREPs*, no *RC*	Low NRL in hi traffics, Fig.3.7.a.	Increased E2ED, Fig.3.5.a.	Low delay Fig.3.5.b.	Increased NRL Fig.3.7.b.
Link state	**FSR**	*GF* technique	Low ANRL, Fig.3.8.a.	Increased NRL, Fig.3.6.a.	Keep low NRL, Fig.3.8.b.	Increased E2ED, Fig.3.6.b.
	OLSR	*MPR* mechanism	Least delay, Fig.3.6.a.	Highest NRL, Fig.3.8.a.	Low delay Fig.3.6.b.	Increased NRL Fig.3.8.b.

all populations and less number of data flows. On the other hand, as, these *MPRs* are calculated through *TC* and *HELLO messages* containing link state information, so, OLSR suffers more ANRL.

3.7 Flooding in reactive protocols

A flooding algorithm is used for exchanging the topological information with every part of a network. In flooding each node can act both as a source and as a router. Each node broadcasts route information to all of its neighbors until destination is reached. This repeated broadcast results in the reception of a particular message by all nodes in the network.

3.7.1 Plain Flooding

In PF [35], on its reception, a control packet is (re)transmitted by all nodes, in the network (except destination). Thus, for N nodes in the network, $N-1$ transmissions are required for a routing packet, when an optimal value is reached for an average path length of the network (L). Since, $\lambda_t N$ data packets are generated each second, the additional bandwidth required for transmission of all these packets is:

$$(N-1-L)\lambda_t N \ bps \tag{3.3}$$

3.7.2 Super Flooding

In SF algorithms, any node retransmits a route request packet if it receives a shorter path than the previously retransmitted route request. This procedure “guarantees” that the shortest path is ultimately most favorable. SF does not acquire more overhead than PF, where the route request is retransmitted by each node on its reception. SF overhead is almost less than the half of PF overhead, and it reduces the data traffic overhead by 60 percent.

3.7.3 MPR Flooding

A node in a network selects a random subset of its 1-hop symmetric neighbors for forwarding a message. This subset is referred as Multi-point Relay *(MPR)* set, and it covers all the nodes that are two hops away from this node. The *MPR* set exchanges routing information by the HELLO messages with both one hop and two hop symmetric neighbors. The neighbor nodes which select an MPR are called 'MPR Selector set'. Unlike plain flooding, in MPR flooding, only MPRs broadcast the routing information that results in controlled routing overhead.

3.7.4 Blind Flooding

A straightforward approach for broadcasting as flooding technique is blind flooding, in which each node is required to rebroadcast the packet whenever it receives the packet for the first time. Blind flooding can cause the broadcast storm problem by generating the redundant transmissions.

Each routing protocol has to pay some cost for the routing overhead generated because of flooding. Inspiring from [36], Saleem *et al.* [37] extended their previous work [38] and proposed the equation (3.4) for calculating the par packet cost (C_p) to be paid for blind flooding.

$$C_p = \begin{cases} P_S d_{avg} & if\ h = 1 \\ P_S d_{avg} + d_{avg} \sum_{i=1}^{h-1} (P_S)^{i+1} \prod_{j=1}^{i} d_f[j] & otherwise \end{cases} \tag{3.4}$$

The number of directly connected or adjacent neighbor nodes within a network for a node is known as *degree* of that node. An isolated node is a node which has zero *degree.* In eq. 3.4, h is the number of hops, d_{avg} is the average degree of node (in Fig. 3.9.a, the average degree is 3, in Fig. 3.9.b, d_{avg} is 2, and for Fig. 3.9.c, it is 3 . $d_f[j]$ is the expected forward degree of a node at the j^{th} hop. Expected forward degree of a node is the average number of neighbors of that node which forward a received RREQ with probability of broadcasting (P_S) [38]. Forward degrees ($d_f[j]$) for Fig. 3.9.a are $4, 2$, for Fig. 3.9.b are $4, 3, 2$ and, for Fig. 3.9.c these are $3, 3$.

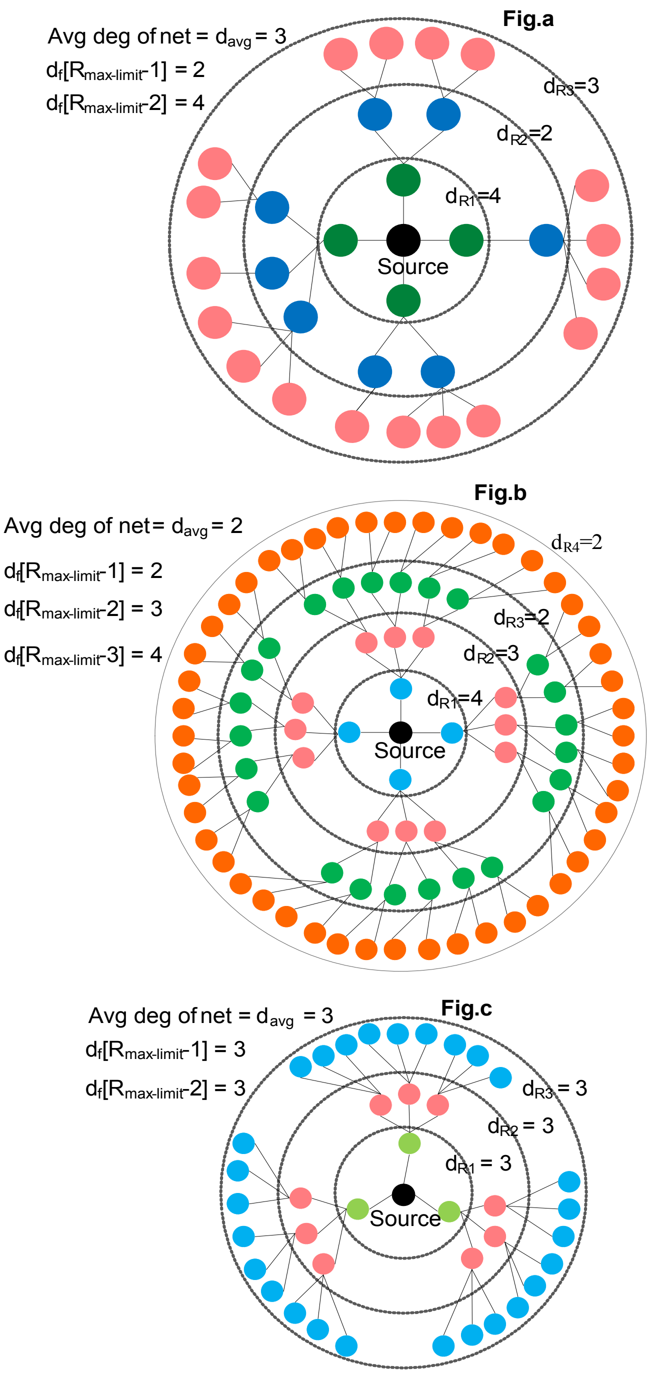


Figure 3.9: Average and Forwarding Degrees of nodes and rings for blind forwarding

In Fig.3.9, there are three different topologies in a network. Eq.3.4 provides an approximation of the packet cost for blind flooding. According to eq.3.4, rings in a network are assigned different degrees with respect to source node. In Fig.3.9.a, all rings as well as all nodes in the same ring have different degrees, in Fig.3.9.b the nodes in the same ring have the same degree, whereas, different degrees of ring in the network. The approximation cost of both of these topologies can be calculated by eq.3.4, while this equation provides exact measurement for routing overhead cost when the degree of the nodes and degree of the rings becomes the same as in Fig.3.9.c.

3.8 Optimization of flooding using Expanding Ring Search Algorithm

Eq.3.4 gives the approximation cost paid by the protocol per packet for route discovery using blind flooding. There are many optimizations to control the routing overhead. Expanding Ring Search (ERS) [39] is one of the optimization techniques, as shown in Fig.3.10. It is adopted by AODV, DSR and DYMO. In ERS, the flooding is controlled by the Time To Live (TTL) values to limit the broadcast.

The process of finding a destination by means of ERS is shown in Fig.3.10. In ERS, source node first broadcasts a RREQ with setting the TTL value to 1. If the reply is not received during a Unit Time (UT)then the RREQ packet is broadcasted by incrementing the TTL value by source node ($TTL = TTL + 1$). Meanwhile, for the successive transmissions of broadcast rings by incrementing the TTL, the waiting time becomes double from the previous ring time ($\tau = \tau + 2$). This process is repeated again and again until the route request finds the desired destination within expanding rate limit or reaches up to the last limit of ERS broadcast.

As, ERS uses blind flooding for broadcasting, so its routing cost can be calculated from eq. 3.4. In the case of ERS, h is replaced by the TTL value in a ring. The $C_{E-R_i}(TTL)$ is the cost of any ring, R_i that generates RREP(s) and the ring R_i is called R_{rrep} and it can disseminate up to the maximum limit R_{max_limit} resulting in either successful or unsuccessful RD; i. e; $R_i \backslash R_i \in R_{rrep} \vee R_i \in R_{max\_limit}$

If P_r is the probability with which a node forwards a RREQ to its neighbors, then P_S becomes P_r in equation.3.5. Here, d_{avg} is the average degree of a node and $d_f[j]$ is the

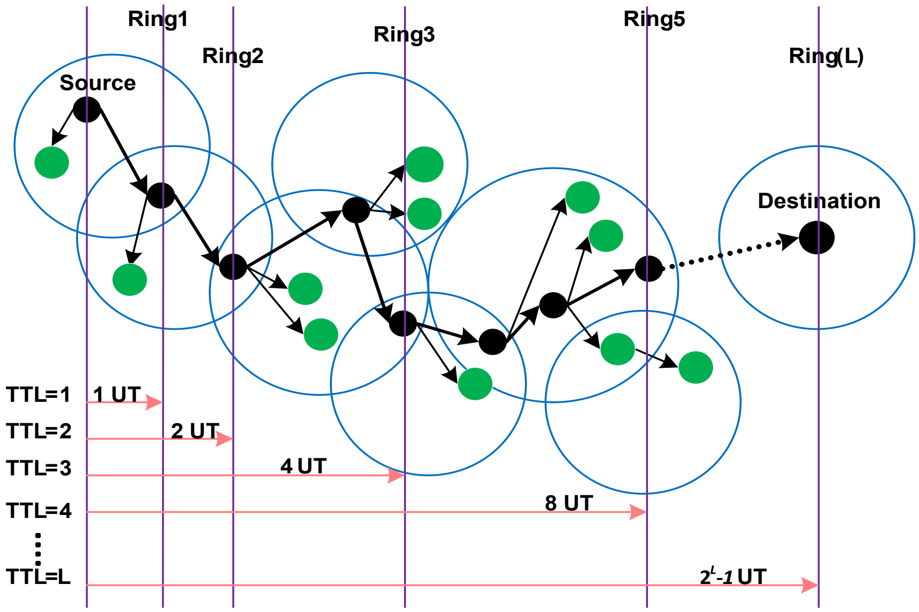


Figure 3.10: Basic Expanding Ring Search Algorithm

expected forward degree of a ring same as that of equation (3.4). $C_{E-R_i}(TTL)$ can be calculated as:

$$C_{E-R_i} = \begin{cases} P_r \times d_{avg} & if\, TTL(R_i) = 1 \\ \{P_r d_{avg} + d_{avg} \displaystyle\sum_{TTL=1}^{TTL(R_i)-1} (P_r)^{TTL+1} \prod_{j=1}^{TTL} d_f[j]\} & otherwise \\ \qquad \{R_i \backslash R_i \in R_i \rightarrow R_{rrep} \vee R_i \in R_i \rightarrow R_{max\_limit}\} & \end{cases} \qquad (3.5)$$

RD using ERS requires broadcast inside the rings by incrementing TTL values relative to the previous TTL value. In ERS, gradual growth of broadcasting ring takes place to reduce the chances of flooding in the entire network that results in formation of different numbers of rings for different broadcasting levels. The collective routing cost of these expanding rings during RD process C_{E-RD} can be computed as:

$$C_{E-RD} = \begin{cases} \displaystyle\sum_{R_i=1}^{R_{max\_limit}} (C_{E-R_i})_{R_i} & if\, no\, RREP\, received \\ C_{E-R_{rrep}} & if\, TTL(R_{rrep}) = 1 \\ \displaystyle\sum_{R_i=1}^{R_{rrep}} (C_{E-R_i})_{R_i} & otherwise \\ \qquad \{R_{rrep} = 1, 2, 3, ..., max\_limit\} & \end{cases} \tag{3.6}$$

3.9 Modeling the cost paid by reactive protocols

A reactive protocol, rp has to pay some cost in the form of consumed energy per packet, $C_E^{(rp)}$ and in the form of time spent per packet, $C_T^{(rp)}$ to encounter the topological changes during the varying number of nodes and traffic rates. In [40], the authors have expressed this cost by the following equation:

$$C_{total}^{(rp)} = C_E^{(rp)} \times C_T^{(rp)} \tag{3.7}$$

The pre-defined constants used by these protocols and the variables, we have defined for the modeling, are given in tables.3.2 and 3.3 along with their description and values.

3.9.1 Cost of Energy Consumption

Each reactive protocol, rp performs RD and RM processes. So, we define the cost to be paid for energy consumption during RD and RM processes; $C_E^{(rp)}$:

$$C_E^{(rp)} = C_{E-RD}^{(rp)} + C_{E-RM}^{(rp)} \tag{3.8}$$

$C_E^{(rp)}$ is different for each reactive protocol due to different routing strategies. The multiple routes in RC reduce the routing overhead with the help of *grat. RREPs* and PS in DSR. In AODV, route length is shortened by *grat. RREPs* to reduce the cost of RD

Table 3.2: Pre-defined Constants used by Reactive Protocols

Constant	**Used by**	**Value(s)**
HELLO_INTERVAL	AODV, DYMO	1000ms
LOCAL_ADD_TTL	AODV	2
MIN_REPAIR_TTL	AODV	Last known hop-count
NET_DIAMETER	AODV, DYMO	AODV=35, DYMO=10
TTL_START	AODV, DYMO	2 for HELLO msgs, 1 for MAC layer ack.
TTL_INCREMENT	AODV, DYMO	2
TTL_THRESHOLD	AODV, DYMO	7
NODE_TRAVERSAL_TIME	AODV, DYMO	40ms
NET_TRAVERSAL_TIME	AODV, DYMO	AODV=5.6s, DYMO=1.92s
RREQ_RATE_LIMIT	AODV, DYMO	10
RREQ_RETRIES	AODV	2
RREQ_TRIES	DYMO	3
TIME_OUT_BUFFER	AODV, DYMO	2
DiscoveryHopLimit	DSR	255 hops
MaxMainRexmt	DSR	2 retransmision
RouteCacheTimeout	DSR	300s
NonpropRequestTimeout	DSR	30ms

process while successful LLR probability; P_S^{LLR} avoids the re-initiation of RD process as shown in the flow chart of reactive protocols in Fig. 3.1.

Energy Consumed for RD

AODV, DSR and DYMO use ERS for RD by broadcasting the RREQ messages from the source node. A source node may receive RREPs from the nodes that contain alternate (short) route for the desired destination. These replies are only used in AODV and DSR and are known as *grat. RREPs*. The destination RREPs are generated by the destination itself (destination RREPs are generated in all the three reactive protocols).

Eq.3.9 gives the cost to be paid for RREQ packets as well as the cost for RREPs produced during RD. n_{rrep} notation is used for node(s) generating the RREP from R_{rrep}.

Table 3.3: Variables defined for this work

Variables	**Description**
$P_{vr}^{(DSR)}$	Probability of valid routes in Route Cache
P_{S}^{LLR}	Probability of successful local link repair
$P_{S}^{(PS)}$	Probability of successful packet salvaging
$P_{S}^{(rp)}$	Broadcasting probability
d_{avg}	Average degree of a node
h	The number of hops
C_p	Routing overhead in term of packet cost
$d_f[j]$	Expected forward degree at j hops away from the source
C_{E-RD}	The energy cost of Route Discovery process
C_{T-RD}	The time cost of Route Discovery process
P_r	Stochastic forwarding
$TTL(R_i)$	Number of hops (TTL) in i_{th} ring
R_{rrep}	Ring that generates route replies
R_{max_limit}	Maximum allowed ring for route discovery
$R_{threshold}$	Ring with threshold node
$R_{netdiameter}$	Value used to disseminate in entire network
C_{E-R_i}	Cost of Energy for any ring i
C_{E-RD}	Cost of Energy Consumption for RD
C_{E-RM}	Cost of Energy Consumption for RM
C_{T-RD}	Cost of Time Consumption for RD
C_{T-RM}	Cost of Time Consumption for RM
$C_{E-LLR}^{(AODV)}$	Cost of Energy Consumption for LLR

$$C_{E-RD}^{(rp)} = \begin{cases} \sum_{R_i=1}^{R_{max\_limit}} (C_{E-R_i})_{R_i} & if\, no\, RREP\, received \\ C_{E-R_{rrep}} + \sum_{n=1}^{n_{rrep}} (RREP)_n & if\, TTL(R_{rrep}) = 1 \\ \sum_{R_i=1}^{R_{rrep}} (C_{E-R_i})_{R_i} + \sum_{n=1}^{n_{rrep}} (RREP)_n & otherwise \\ \qquad \{R_{rrep} = 1, 2, 3, ..., max\_limit\} & \end{cases} \tag{3.9}$$

The generation of RREP(s) in AODV and DSR is also due to the valid routes in RT or RC, so the ring R_i value for DSR and AODV is less than DYMO. As, grat. RREPs are absent in DYMO. $rrep$ in R_{rrep} can be written as: $R_{rrep} = 1, 2, 3, ..., max\_limit$. Whereas, d_{rrep} represents RREP generated by destination and g_{rrep} shows the grat. RREPs. The value of R_{rrep} depends upon the strategies of a protocol.

Table 3.4: ttl_values and $waiting\_time(ms)$

Ring	ttl_values			$waiting\_time(ms)$		
	AODV	**DYMO**	**DSR**	**AODV**	**DYMO**	**DSR**
R_1	2	2	1	320	320	30
R_2	4	4	2	480	480	60
R_3	6	6	4	640	640	120
R_4	35	10	8	2600	960	240
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
R_{max}	70	20	255	5600	1920	7680

Energy Consumed for RM

In RM process, different protocols pay different costs for link monitoring, $C_{E-l-mon}$, and also there are different costs for supplementary maintenance strategy in case of link breakage, C_{E-LLR} for AODV, C_{E-PS} for DSR, and DYMO does not use any mechanism. Following equations give RM cost for three protocols:

$$C_{E-RM}^{(AODV)} = C_{E-l-mon} + C_{E-LLR} + \sum_{z=0}^{n} (RERR)_z \tag{3.10}$$

$$C_{E-RM}^{(AODV)} = \frac{\tau_{link-in-use}}{\tau_{HELLO\_INTERVAL}} \times N_{hops-in-route} + P_r d_{avg} + d_{avg} \sum_{TTL=1}^{TTL(R_{LLR})-1} (P_r)^{TTL+1} \prod_{j=1}^{TTL} d_f[j] + \sum_{z=0}^{n} (RERR)_z$$

$$C_{E-RM}^{(DSR)} = C_{E-PS} + \sum_{z=0}^{n} (RERR)_z \tag{3.11}$$

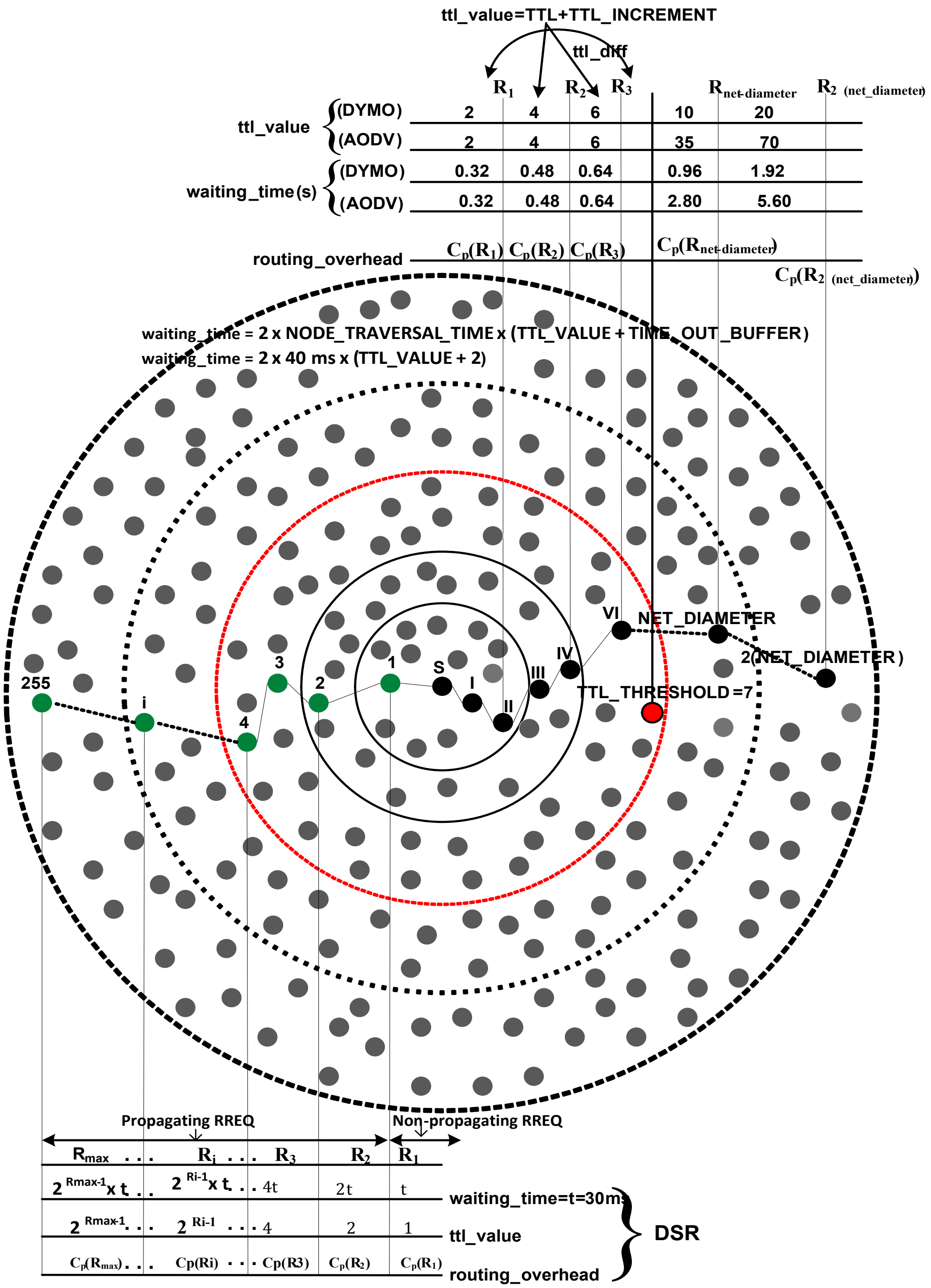


Figure 3.11: Expanding Ring Search Algorithm used by AODV, DSR, DYMO

$$C_{E-RM}^{(DSR)} = \sum_{k=n_{BLB}}^{n_{PS}} (RREQ)_k + \sum_{z=0}^{n} (RERR)_z \tag{3.12}$$

$$C_{E-RM}^{(DYMO)} = C_{E-l-mon} + \sum_{z=0}^{n} (RERR)_z \tag{3.13}$$

$$C_{E-RM}^{(DYMO)} = \frac{\tau_{link-in-use}}{\tau_{HELLO\_INTERVAL}} \times N_{hops-in-route} + \sum_{z=0}^{n} (RERR)_z \tag{3.14}$$

Where, n_{BLB} is the node before link break and n_{PS} may be any node from source to n_{BLB}. In wireless environment, there are frequent link breakages that lead to link failures. As a result, the routes become ineffective. The link breakage is detected by different protocols by their own strategies. DYMO and AODV generate HELLO messages to check the connectivity of active routes, while DSR gets the link level feedback from link layer. This cost depends on path time; i.e., a path in use and length of the path (in terms of hops) and the value of $HELLO\_INTERVAL$ constant.

Broadcast need to send n number of RERRs depending upon different situations for different protocols:

In DYMO, the link breakage causes the broadcasting of RERR messages.

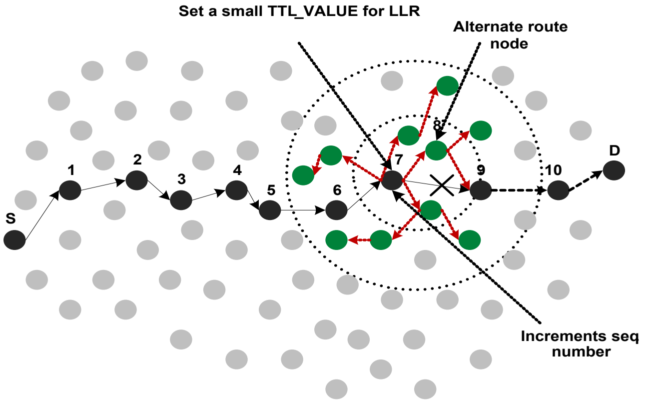
When the probability of successful LLR; P_S^{LLR} becomes zero then it leads to the dissemination of RERRs in AODV.

On the other hand, DSR piggybacks RERR messages along with next RREQs in the case of route re-discovery process, while these RERR messages are broadcasted in the case of $P_S^{(PS)}$; successful probability of PS.

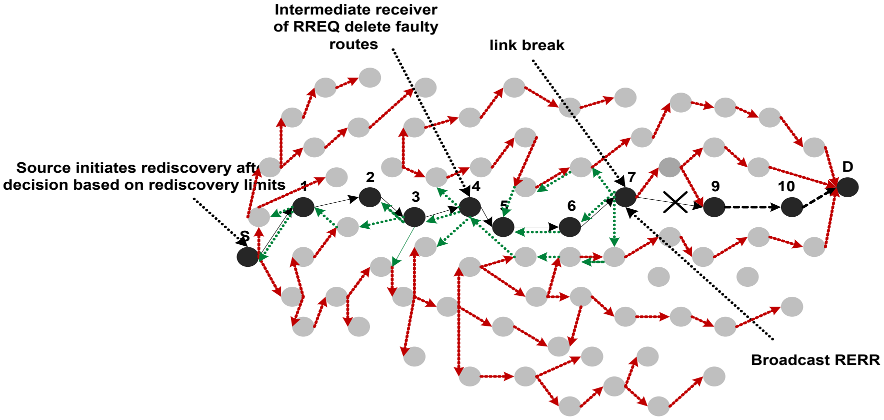
In Fig.3.12, there are three different scenarios for reactive protocols describing the route repair mechanism after detection of route failure because of link breakage. The most simple mechanism in Fig.3.12.b describes that RD re-initiation process takes place under the limited retries constraint for route re-discovery process:

$RREQ\_RETRIES = 3$ in DYMO,

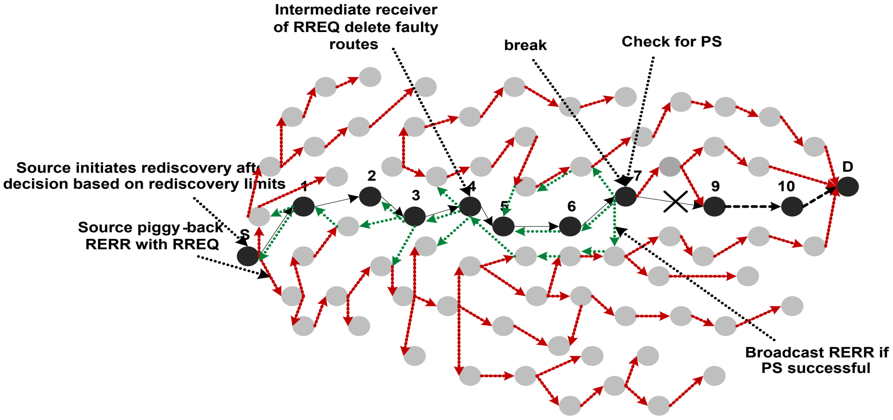
$RREQ\_TRIES = 2$ in AODV, and



(a) AODV Local Link Repair



(b) AODV_DYMO Route Repair



(c) DSR Route Repair

Figure 3.12: Route Re-discovery in Reactive Protocols

$MaxMainRexmt$ in DSR.

In AODV after unsuccessful RD and normally in DYMO, the RERR messages are broadcasted by the node which detects the link break and route rediscovery process is started through source node. Fig.3.12.a clearly demonstrates the importance of LLR in AODV. If it becomes successful in a dense network then it saves the energy consumed during route re-discovery. On the other hand, if LLR becomes unsuccessful then energy cost is increased by re-initiating RD process after performing LLR strategy, as depicted in Fig.3.12.b.

DSR's *PS* technique can reduce both the energy and time cost to be paid by a reactive protocol by diminishing the route re-discovery. In the case of successful *PS*, the RERR messages are broadcasted to neighbors for the deletion of useless routes. Whereas, the absence of alternate route(s) in RC leads to the failure of *PS*. In this situation, RERR messages are to be sent through piggybacking them in the next RREQ messages during route re-discovery process .

Cost of LLR in AODV is given in the following equation.

$$C_{E-LLR}^{(AODV)} = P_r d_{avg} + d_{avg} \sum_{TTL=1}^{TTL(R_{LLR})-1} (P_r)^{TTL+1} \prod_{j=1}^{TTL} d_f[j] \tag{3.15}$$

Here, R_{LLR} represents the ring that limits the LLR activity. The TTL value for R_{LLR} is calculated with $LOCAL\_ADD\_TTL(=2)$ and MIN_REPAIR_TTL (it is the last known hop-count to the destination). The per packet cost of LLR; $C_{E-LLR}^{(AODV)}$ depends upon the TTL value of R_{LLR}. In large networks, the successful LLR process is more useful, because the chances of route re-discovery can be reduced which utilizes more bandwidth space as compared to LLR (as small values of TTL are set for LLR). The value $TTL(R_{LLR})$ is obtained from the equation given below:

$$max(MIN\_REPAIR\_TTL, 0.5 \times \#hops) + LOCAL\_ADD\_TTL \tag{3.16}$$

Where $\#hops$ is the number of hops to the sender of the currently undeliverable data packet. Thus, local repair attempts will often be imperceptible to the originating node, and will always have $TTL >= MIN\_REPAIR\_TTL + LOCAL\_ADD\_TTL$.

3.9.2 Cost of Time Consumption

The cost of end-to-end path calculation time $C_T^{(rp)}$ in reactive protocols depends upon route discovery time $C_{T-RD}^{(rp)}$ and $C_{T-RM}^{(rp)}$.

$$C_T^{(rp)} = C_{T-RD}^{(rp)} + C_{T-RM}^{(rp)} \tag{3.17}$$

Time Consumed for RD by DSR

τ is constant time initially used for non-propagating RREQ ($NonpropRequestTimeout$) and its value is $30ms$. R_{max_limit} is the maximum ring size and it depends on the buffer time as well as the maximum allowed broadcasting during propagating RREQ ($DiscoveryHopLimit =$ 255). The binary exponential back-off is associated with each propagating ring.

$$C_{T-RD}^{(DSR)} = \begin{cases} \tau & if\, R_{rrep} = 1 \\ \displaystyle\sum_{R_i=1}^{R_{max\_limit}} 2^{R_i-1} \times \tau & if\, no\, RREP\, received \\ \displaystyle\sum_{R_i=1}^{R_{rrep}} 2^{R_i-1} \times \tau & otherwise \end{cases} \tag{3.18}$$

Time Consumed for RD by AODV and DYMO

Both in AODV and DYMO, firstly, the TTL_VALUE in IP header is set to TTL_START (=1 in the case of link layer feedback otherwise =2) then it is increased by $TTL\_INCREMENT(=$ 2) up to $TTL\_THRESHOLD(= 7)$ [18]. After $TTL\_THRESHOLD$, the TTL_VALUE is set to the $NET\_DIAMETER(= 35)$. For dissemination in the entire network the TTL_START and $TTL\_INCREMENT$ both are set to $NET\_DIAMETER$ (for AODV = 35 [18] and for DYMO = 10 [21]). Moreover, the maximum RREQ tries are 3 for DYMO [21], and maximum retries are 2 for AODV. The $RREQ\_TIME$ is set to $2 \times$ $NET\_TRAVERSAL\_TIME$ (whereas, $NET\_TRAVERSAL\_TIME = 2 \times NODE\_TRAVERSAL\_T$ $NET\_DIAMETER$).

$$C_{T-RD}^{(AODV,\,DYMO)} = \begin{cases} \sum\limits_{R_i=1}^{R_{max\_limit}} \tau_1(TTL(R_i) + \tau_2) & if\,no\,RREP\,received \\ \sum\limits_{R_i=1}^{R_{rrep}} \tau_1(TTL(R_i) + \tau_2) & otherwise \end{cases} \tag{3.19}$$

There are two possibilities for AODV and DYMO that either the RD process becomes successful in threshold rings $R_{thereshold}$ or RD process needs to disseminate the request in the whole network ($R_{netdiameter}$). Whereas, $TTL(R_{thrshold})$ and $TTL(R_{netdiameter})$ represent the TTL value in a ring which generates the RREP(s) either among the rings within $THERESHOLD$ limit or from the entire network ($NET\_DIMETER$ value is used to broadcast RREQ in the entire network).

Time Consumed for RM in AODV

AODV starts LLR process after noticing a link failure. C_{T-LLR} gives the time cost of LLR that depends upon the TTL value of the ring; $LLR(R_{LLR})$ in which LLR is to be performed. In the case of LLR failure, AODV disseminates RERR messages. $\tau_{recv-RERR}$ represents the time which is spent to reach RERR message from the node detecting the link failure to the originator node. $C_{T-re-RD}$ cost is to be paid to start route re-discovery based on the value $RREQ\_RETRIES(= 2)$.

$$C_{T-RM}^{(AODV)} = \begin{cases} \sum\limits_{R_i=1}^{R_{LLR}} \tau_1(TTL(R_i) + \tau_2) & if\,LLR\,is\,successful \\ \sum\limits_{R_i=1}^{R_{LLR}} \tau_1(TTL(R_i) + \tau_2) + \tau_{recv-RERR} & if\,LLR\,fails,\,RREQ\_RETRIES\,expi \\ \sum\limits_{R_i=1}^{R_{LLR}} \tau_1(TTL(R_i) + \tau_2) + \tau_{recv-RERR} + C_{T-re-RD} & otherwise \end{cases}$$

where,

$$C_{T-re-RD} = C_{T-LLR} \tag{3.21}$$

Time Consumed for RM in DSR

After detecting a link failure, time τ_{PS} is utilized to check alternative routes in *RC's* of intermediate nodes (from a node which detects link failure to a node having alternate route for this broken link; n_{alt-r}. This τ_{PS} value is higher; if node containing alternative route; n_{alt-r} is nearest to the node which detects link breakage. In the case of failure of PS or in the case of presence of alternative route in RC of the originator node, τ_{PS} attains a maximum value and is consumed by all intermediate remaining nodes $n - irr$ in a route (from a node that detects link break up to the originator of this broken route).

$$C_{T-RM}^{(DSR)} = \begin{cases} \sum_{k=n_{BLB}}^{n_{PS}} \tau_k(PS) & if\, PS\, is\, successful \\ \sum_{k=n_{BLB}}^{n_{originator}} \tau_k(PS) + C_{T-re-RD} & otherwise \end{cases} \tag{3.22}$$

where, $C_{T-re-RD} = C_{T-RD}$

Time Consumed for RM in DYMO

A RERR message is broadcasted by the node that detects link break. After a time $\tau_{recv-RERR}$, which is consumed for receiving RERR message by the source node, source node initiates route rediscovery; $C_{T-re-RD}$ that is based on $RREQ\_RETRIES (= 3)$ constraint.

$$C_{T-RM}^{(DYMO)} = \begin{cases} \tau_{recv-RERR} & if\, RREQ\_TRIES\, expires \\ \tau_{recv-RERR} + C_{T-re-RD} + & otherwise \end{cases} \tag{3.23}$$

Time TRAD-OFFs of PS and LLR

In *PS* of DSR, the first checking of *RC* of the intermediate nodes for alternate route(s) consumes more time. In the case of successful *PS*, the time can be reduced as compared to time consumption for route re-discovery from source for end-to-end path calculation. On the other hand, the *PS* checking time adds up with route re-discovery time in the case of failure.

The same case is with *LLR* in AODV. The success of the process lessens the end-to-end path time because route re-rediscovery process is not initialized. While, *LLR* increase the path length in case of unsuccessful repair, because the repair time is also added with re-discovery time. As, LLR is performed by broadcasting a small TTL value, so, R_{LLR} consumes some time during repair. The time cost of LLR in AODV $C_{T-LLR}^{(AODV)}$ can be calculated as:

$$C_{T-LLR}^{(AODV)} = \sum_{R_i=1}^{LLR} \tau_1(TTL(R_i) + \tau_2) \tag{3.24}$$

3.10 Conclusion

In this chapter, a detailed analysis evaluation and then comparison of both reactive and proactive protocols are carried-out. We have taken scalability and traffic loads into account. The performance parameters, throughput, end-to-end delay, and normalized routing load are used for measuring the capabilities of the protocols in different scenarios. The total overhead attained by a routing protocol consists of two parts; overhead in the form control traffic generated by the control packets and overhead in the form of data traffic forwarded through routes of non-optimal length. Increased routing overhead and delay are major issues of concern to be resolved by the routing protocols in wireless environment. We have also modeled the cost paid by the reactive protocols for the generated routing overhead. The cost consists of the energy consumed and time spent per packet for route discovery and route maintenance process.

In dense networks, the optimization of retransmissions results better performance of a protocol. While the reduction of network bandwidth utilization is more useful when data flows are increased. Finally, we analyze that AODV because of distance vector dissemination reduces bandwidth consumption, and LLR reduces the retransmission attempts making this protocol better among reactive protocols. OLSR due to reduction of retransmissions through MPRs is more suitable for dense networks, and FSR due to reduction of routing overhead performs better in high data traffic rates.

Chapter 4

ETX-based Routing Metrics

Contents

4.1 Introduction

On a wireless link, the number of link layer transmissions of a packet is an appealing cost metric because minimizing the total number of transmissions (and retransmissions) maximizes the throughput of an individual link then overall network. ETX proposed in [41], [42], measures MAC transmissions and retransmissions to recover from frame losses since the link level retransmissions depend only on the link level packet errors caused by channel issues. ETX of a wireless link is the estimated average number of transmissions of data frames and ACK frames necessary for the successful transmission of a packet [43]. A node derives ETX by estimating the frame loss ratio at the link l to each of its neighbors in the forward direction as p_{lf}, and in the reverse direction as p_{lr} transmitting broadcast probe packets (which are not retransmitted) at the link layer once every second as:

$$ETX_l = \frac{1}{(1 - p_{lf}) \times (1 - p_{lr})} \tag{4.1}$$

Alternatively, ETX of the link is the inverse of the probability of ”successful packet delivery” or ”link reliability”:

$$ETX_l = \frac{1}{(p_{lsf}) \times (p_{lsr})} = \frac{1}{reliability(l)} \tag{4.2}$$

If we increase the frequency of ETX measurements and change the optimum paths accordingly more frequently, it involves significant amount of overhead in the network. It has been shown that the link with a lower ETX metric may in fact lead to a higher observed loss rate at the transport layer. Because good link-layer protocols do not retransmit lost packets forever and give up after a threshold number of attempts. The losses occurring in the form of bursts cause to pick the link in the middle of a burst-error situation, which is bad even with a lower ETX.

Consider for example, the Fig.4.1, which illustrates the packet delivery ratios taken from four distinct links in the Roofnet wireless mesh network [44]. Each of these four links has an ETX around 2 during the testing period. Therefore, if ETX is taken as the metric for quality, these four links are identical. On the other hand, the sample variances of the delivery ratios are quite different for these links, i.e., these wireless links have similar long term average behaviors, even though their short-term behaviors are quite different [45].

Pithily, ETX does improve the throughput of a wireless network (with less mobility) when compared to hop count metric but it does not track the variations on the channel at short time scales due to potential route instability [46]. Table 4.1 lists the performances over ETX, design goal and experimented platforms of the ETX based metrics.

4.2 Study of ETX based link metrics

This section is dedicated for the discussion of the up-to-date metrics that are based on ETX.

4.2.1 Modified ETX (mETX) and Effective Number of Transmissions (ENT)

In almost all kinds of wireless networks, due to the fast link-quality variation, the metrics based on a time-window interval, such as ETX, ETT, WCETT, MIC, MCR, iAWARE, etc., may not follow the link quality variations and/or may produce prohibitive control overhead. To cope with the situation, mETX and ENT were proposed in [45], which are aware of the probe size, therefore, the inclusion of the data rate is trivial for them. Along with the link-quality average values, these metrics consider the standard deviation to project physical-layer variations.

A. Modified ETX (mETX)

Presence of channel variability in ETX became the reason to design mETX. The difference between mETX and ETX is: rather than considering probe losses, mETX works at the bit level. The mETX metric computes the bit error probability using the position of the corrupted bit in the probe and the dependence of these bit errors throughout successive transmissions. This is possible because probes are composed by a previously known sequence of bits. The variability of the link is modeled using the statistics of this stochastic process. Then, the mean number of transmissions is analytically calculated and the results show that it can be closely approximated with the statistics of the bit error probability, summed over a packet duration. For mETX, the critical time scale for the link variability is the transmission time of a single packet including all its retransmissions. mETX is defined in eq.(4.3) with μ being the estimated average packet loss ratio of a link and σ^2 the variance of this value. Like ETX, mETX is additive over concatenated links.

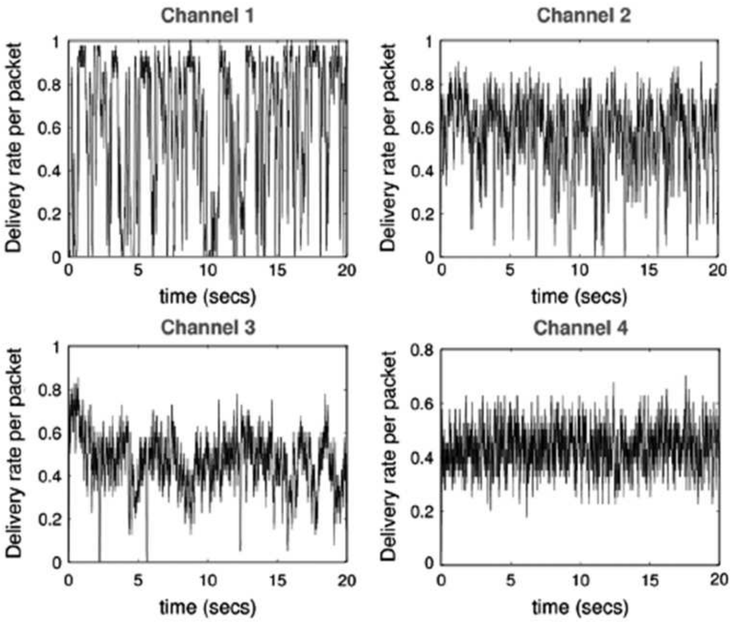


Figure 4.1: Roofnet, Packet delivery rate for four distinct links

$$mETX = exp\,(\mu + \frac{\sigma^2}{2}) \tag{4.3}$$

μ means the impact of slowly varying and static components, like shadowing, slow fading in the channel and σ^2 shows the impact of relatively rapid channel variations, as fading, interference, etc. which the term μ_Σ (and hence the ETX) cannot track. μ and σ^2 are estimated by the bit level information, as counting only the packet losses is not sufficient; thus, parameters μ and σ^2 are estimated by considering the number of erred bits in each probe packet. Complexity of "channel estimation" is the main disadvantage of the mETX, as:

(1) probe packets are to be processed at the bit level raising energy consumption issue in wireless sensor networks (which may not be an issue for wireless mesh networks due to their abundance of processing power),

(2) σ^2 increases with increased estimation error. A link's high mETX is due to high channel variability and estimation error which results a better link to be malformed. mETX can be adapted like ETX easily for those wireless links which provide bit rate adaptation by normalizing the metric according to the transmission rate.

B. Effective Number of Transmissions (ENT)

The upper-layer protocols, as transmission Control protocol (TCP), Sequenced Packet exchange (SPX), etc. have a limit to tolerate (re)transmissions. This issue caused ENT to be proposed. ENT therefore, broadcasts probes, limits routes to an acceptable number of (re)transmissions according to the requirements of upper-layer. It measures the number of successive retransmissions per link considering the variance to find a path that achieves high network capacity while ensuring that the end-to-end packet loss rate visible to higher layers (such as TCP) does not exceed a specified value but this may not be sufficient, as it may involve links with high loss rates. ETX and mETX metrics usually select the links which do not obey the transmission threshold required by the algorithms working at higher layers.

Let M be the threshold number of retransmissions (specified by higher layers), P_{al} (actual probability of a packet loss), using a large deviations approach can be defined as:

$$P_{al} = exp\,[\frac{-(logM - \mu)}{2\sigma^2}] \tag{4.4}$$

G be the temporal diversity gain for a wireless link:

$$G = \frac{-logP_{al}}{logM} \tag{4.5}$$

which specifies the desired loss probability P_{dl}. Now, ENT can be defined as:

$$ENT = exp\,(\mu + 2G\sigma^2) \tag{4.6}$$

If higher layer does not specify any loss probability constraint, i.e.,$G = 0$, then for the given P_{dl}, $2G\sigma^2 \leq logM$, we left with $\mu \leq logM$. If $G > 0$ then there must be efficient resources for the network to put an amount $2G\sigma^2$ (which is directly related to the variability of the channel, σ^2 and strictness of the loss requirement, G). This interpretation of ENT is analogous to the notion of effective bandwidth, which was developed to model variable traffic sources in queuing networks. Indeed, ENT can be interpreted as the effective bandwidth of the discrete stochastic process, the number of transmissions.

Differences between the two include: (1) an extra degree of freedom due to the factor $2G$ in ENT. Indeed (mETX is the ENT evaluated at 1/4), (2) ENT is not additive as ETX or mETX. Similarities between the two include: (1) a by-product of ENT reduces the packet loss ratio observed by higher-layer protocols, after any link-layer retransmissions are done, (2) they have same channel estimation procedure. Main feature of ENT is that it can be calibrated. It is useful to have a degree of freedom for the necessary adjustments derivations in [3] are based on certain assumptions, which can be partly violated in different platforms and environments. Both of drawback of mETX metric are valid for the ENT as well.

4.2.2 Expected Transmission Time (ETT)

ETT is the time a data packet needs to be successfully transmitted to each neighbor. To overcome ETX's shortcomings: (1) it broadcasts at the network basic rate, (2) its probes are smaller than data packets, ETT [43] adjusts ETX to different PHY rates and data-packet sizes. Two approaches to compute the bandwidth of link l,B_l are:

Eq.(4.1) from [47] can be re-written as:

$$ETT_l = ETX_l \times t \tag{4.7}$$

$$ETT_l = ETX_l \times \frac{S_F}{B_l} \tag{4.8}$$

$$ETT_l = ETX_l \times \frac{S_F}{\frac{S_L}{T_S - T_L}} \tag{4.9}$$

Where S_F is the data packet of fixed-size, B_l is bandwidth of link l, S_L is data packet of largest-size, $T_S - T_L$ is an interval between the arrivals of two packets. This technique unicasts two packets in sequence, a small one followed by a large one, to estimate the link bandwidth to each neighbor measuring the inter-arrival time period $T_S - T_L$ between the two packets and reporting it back to the sender. The computed bandwidth is the size of the large packet of the sequence divided by the minimum delay received for that link.

Eq.(4.2) from [42], i.e., loss probability is estimated by considering that IEEE 802.11 uses data and ACK frames. Loss rate of data is estimated by broadcasting a number of packets of the same size as data frames, one packet for each data rate defined in IEEE 802.11. Loss rate of ACK frames is estimated by broadcasting small packets, of the same size as ACK frames and sent at the basic rate, which is used for ACKs. ETT may choose a path that only uses one channel, even though a path with more diversified channels has less intra-flow interference and hence higher throughput. Similarly to ETX, the chosen route is the one with the lowest sum of ETT values.

4.2.3 Weighted Cumulative ETT (WCETT)

Basically WCETT is based on ETT and is aware of the loss rate (due to ETX) and the bandwidth of the link. Proposed by [47], WCETT can be used for multi-radio, multi-hop WMN. It proposes ETT which improves on ETX by making use of the data rate in each link. The ETT of a link is defined in eq(4.7). ETT explains the expected MAC transmission time of a packet of a size S over certain link l. Given the presence of multiple channels and intra-flow interfere, WCETT is defined as:

$$WCETT = (1-\beta) + \sum_{i=1}^{n} ETT_l + \beta \times \max_{1\leq j\leq k} X_j \tag{4.10}$$

WCETT is the sum of ETTs of all the links in path p operating on X_j channel j, in a system with total of k orthogonal channels. β is a tunable parameter subject to $0 \leq \beta \leq 1$. WCETT consists of two components: the first component finds the path with the least sum of ETTs; the second accounts for the bottleneck channel dominating the throughput of the total path.

Its advantages include: **(1)** over performing ETT, it explicitly accounts for the intra-flow interference, providing support for multi-radio or multi-channel wireless networks, **(2)** its two weighted components of it substitute the simple summation of ETT and attempt to strike a balance between throughput and delay. It does not capture inter-flow interference compared with Interference Aware Routing Metric (iAWARE). It modifies ETT considering intra-flow interference. This metric is a sum of end-to-end delay and channel diversity. Like Minimum Loss (ML) and unlike ETX and ETT, WCETT is an end-to-end metric because it must consider all channels used along the route to avoid intra-flow interference.

4.2.4 Metric of Interference and Channel-switching (MIC)

WCETT does avoid intra-flow interference but it does not (1) guarantee shortest paths (2) avoid inter-flow interference; which may lead WCETT to select congested routes. MIC [48], [49] tackles these issues by providing the features:

(a) each node estimates inter-flow interference by counting the number of interfering nodes in the neighborhood.

(b) MIC virtual nodes guarantee minimum-cost routes computation.

(c) MIC calculates itself by ETT metric. MIC for a path p is defined as follows:

$$MIC(p) = \frac{1}{N \times min(ETT)} \sum_{i \in p} IRU + \sum_{i \in p} CSC_i \tag{4.11}$$

Where N is the total number of nodes in the network and $min(ETT)$ is the smallest ETT in the network. The two components of MIC, IRU (Interference-aware Resource Usage) is $IRU_l = ETT_l \diamond N_l$ and CSC (Channel Switching Cost) is defined as:

$CSC_i = w_1; ifCH(prev(i)) \neq CH(i)$ and

$CSC_i = w_2; ifCH(prev(i)) = CH(i)$,

where $0 \leq w_1 \leq w_2$ and N is the set of neighbors that interfere with the transmissions on link i. $CH(i)$ represents the channel assigned for node i's transmission and $prev(i)$ represents the previous hop of node i along the path p. MIC takes the inter-flow interference into account. Its disadvantages include: (1) the component, CSC captures intra-flow interference only in two consecutive links. (2) MIC considers interference of a link caused by each interfering node in the neighborhood, counts the amount of interferers on a link only by the position of the interfering nodes no matter whether they are involved in any transmission simultaneously with that link. MIC, therefore, utilizes the measurement of signal power to capture inter-flow and intra-flow interference.

4.2.5 Interference AWARE (iAWARE)

iAWARE considers not only both inter-flow and intra-flow interference and characterized by the physical interference model but also takes link-quality variation into account. This metric uses Signal to Noise Ratio (SNR) and Signal to Interference and Noise Ratio (SINR) to continuously reproduce neighboring interference variations onto routing metrics. The iAWARE metric estimates the average time the medium is busy because of transmissions from each interfering neighbor. Higher the interference, higher the iAWARE value. Thus, unlike mETX and ENT, iAWARE considers intra-flow and inter-flow interference, medium instability, and data-transmission time. In this model [50], a communication between nodes u and $v$on the link $(u \rightarrow v)$ is successful, if the SINR at the receiver v is above a certain threshold. Let $P_u(v)$ denotes the signal strength of a packet from node u to node v. iAWARE's first component, finds paths with least path cost and other finds paths with least intra-flow interference (exploiting channel diversity). Moreover, the introduction of SINR is a great breakthrough for inter-flow interference-aware routing compared with other ETX-based metric, like MIC.

Definition of the link metric iAWARE of a link j as follows:

$$iAware_j = \frac{ETT_j}{IR_j} \tag{4.12}$$

When IR_j for the link j is 1 (no interference), $iAWARE_j$ is simply ETT_j which captures the link loss ratio and packet transmission rate of the link j. ETT_j is weighted with IR_j to capture the interference experienced by the link from its neighbors. A link with low ETT and high IR will have a low iAWARE value. Lower the iAWARE of a link better is the link. We define interference ratio $IR_i(u)$ for a node u in a link $i = (u, v)$, where $IR_i(u)$ $(0 < IR_i(u) \leq 1)$ can be defined as:

$$IR_i(u) = \frac{SINR_i(u)}{SNRi(u)} \tag{4.13}$$

$$SNR_i(u) = \frac{P_u(v)}{N} \tag{4.14}$$

$$SINR_i(u) = \frac{P_u(v)}{[N + \sum_{w \in \eta(u)-v} \tau(w) P_u(w)]} \tag{4.15}$$

Here $\eta(u)$ denotes the set of nodes from which node u can hear (or sense) a packet and $\tau(w)$ is the normalized rate at which node w generates traffic averaged over a period of time. $\tau(w)$ is 1 when node w sends out packets at the full data rate supported. We use $\tau(w)$ to weight the signal strength from an interfering node w as $\tau(w)$ gives the fraction of time node w occupies the channel.

4.2.6 Distribution Based Expected Transmission Count (DBETX)

Through a complete physical channel view and using cross-layer optimizations, Distribution Based Expected Transmission Count (DBETX) is proposed in [51] to improve network performance for varying channels and in the presence of fading.

DBETX's performance over ETX increases with the network density because connectivity increases and more routing options become available. Results show a reduction of up to 26% in the Average Number of Transmissions (ANT) per link and an increase of up to 32% in the end-to-end availability. Using link measurements, DBETX makes the nodes able to: (i) estimate the probability density function (pdf) of the experimented SNIR, (ii) calculate the expected Bit Error Rate (BER) and, as a consequence, the expected packet error rate (PER), (iii) estimate average number of required transmissions in a given link based on the SNIR, (iv) derive the number of required transmissions taking into account the maximum number of MAC-layer retransmissions, (v) penalize lossy links in order to find routes with lower end-to-end loss rates, (vi) reflect the variations of the wireless channel, (vii) to favor links with a lower loss probability (oppositely from [45]). DBETX metric for a link is defined as:

$$DBETX(l) = E[ANT(i)] \times \frac{1}{1 - P_{out_{MAC}}} \tag{4.16}$$

$P_{out_{MAC}}$ is the probability when $P_{suc}(x) < P_{limit}$. Where $ANT(l)$ is given by:

$$ANT(l) = \frac{1}{P_{suc}(x)}; P_{suc} > P_{limit} \tag{4.17}$$

$$ANT(l) = \frac{1}{P_{limit}(x)}; P_{suc} \leq P_{limit} \tag{4.18}$$

MAC layer outage (a condition when current Success Probability (P_{suc}) of a link results in an expected number of retransmissions higher than $MaxRetry$) occurs when the success probability of a link is smaller than the Limit Success Probability (P_{limit}), which is $P_{limit} = \frac{1}{MaxRetry}$. In this situation, there is a high probability that the transmitted packet will be discarded due to an excessive number of retransmissions. ANT function is the expected number of retransmissions on a link considering the value of $MaxRetry$, which is the maximum number of retransmissions allowed by the MAC-layer (For IEEE 802.11, it is 7 in the presence of Request to Send / Clear To send (RTS/CTS) handshake). DBETX's calculation requires the information of actual behavior of wireless link instead of the average behavior. Due to the difference in the working time scales of the different layers, it is impossible to have a complete view of physical medium based on network level, as the events of interest occur at the physical level at milli or microseconds, network level interactions are reduced in order to reduce overhead at a time scale of seconds.

4.2.7 Exclusive Expected Transmission Time (EETT)

In large-scale multi-radio mesh networks (LSMRMNs), most of traffic has much longer paths than in small scale WMNs [52]. When channels are distributed on a long path, EETT selects multi-channel routes with the least interference to maximize the end-to-end throughput. None of existing routing metrics is capable to evaluate two multi-channel paths accurately when the paths are long. So, EETT well considers channel distribution on long paths which however are very critical in LSMRMNs. In order to meet the above mentioned requirement, EETT is used to give a better evaluation of a multi-channel path. For a $N-hop$ path with K channels, on a link l, its Interference Set (IS) is the set of links that interfere with it (a link's IS also includes the link itself). Then this link l's EETT is defined as:

$$EETT_l = \sum_{l \in IS(l)} ETT_l \tag{4.19}$$

Physical interpretation of EETT states that EETT of a link l shows the channel used by link l. Link l may have to wait a longer period for transmission on a channel, if there are more neighboring links on that channel with link l resulting in a path with a larger EETT with more severe interference and needs more time to finish the transmission over all links within the path. EETT reflects the optimality of the channel distribution on a path, as this results in less intra-flow interference. EETT can also embody the inter-flow interference, if $IS(l)$ includes those links which do not belong to the same path with link l. MIC considers the impact of link l on other links, while EETT considers the impact of other links on link l hence EETT is supposed to have better performance since it more accurately reflects the impact of the inter-flow interference.

4.2.8 Expected Data Rate (EDR)

To overcome ETX's key limitation of not taking into account the multi-rate links, ETT was proposed to account for multi-rate links. Transmission Contention Degree, (TCD) in [53] was defined to overcome the limitation of ETX and ETT for making conservative estimates for paths longer than 3-4 hops (as all the cochannel links on a path contend with each other) by incorporating time-sharing effects of MAC. TCD is the average fraction of the time for which the outgoing queue of the transmitter of link l is non-empty. EDR is defined as:

$$EDR_l = \frac{b_l}{ETX_l \times \sum_{i=1}^{n} TCD_l(i)} \tag{4.20}$$

Where b_l is the nominal bit rate of the link l. $\sum_{i=1}^{n} TCD_l(i)$ is used to account for throughput reduction due to equal time-sharing with the contending links provided that all the links have the same nominal bit rate. If links have different nominal bit rates, they receive the same average throughput, but different time-share of the channel failing to capture the bandwidth-sharing mechanism of 802.11 DCF.

4.2.9 Expected Throughput (ETP)

Being proposed in [54], ETP: (1) predicts better routes than ETX and ETT in mesh networks with long paths, they do not make spatial measurements, (2) also measures expected throughput of a link, (3) can easily be implemented in the IBSS mode with minor additions to the beacon message contents, (4) predicts better routes in mesh networks with heterogeneous link rates because ETP captures the bandwidth sharing mechanism of 802.11 DCF more accurately than EDR, ETT, and ETX, as they do not take into account the throughput reduction of fast links due to contention from slow links. (5) ETP is suitable for multi-rate, multi-radio mesh networks.

To state ETP, let link l belongs to path P in the contention domain S_l. $S_l \bigcap P$ is the set of links on path P that contend with link l. r_l be the nominal bit rate of link l. All links have equal number of opportunities for transmission when saturated, as per 802.11 DCF. The expected bandwidth received by each link l is:

$$b_l = \frac{1}{\sum_{j \in S_l \bigcap P} \frac{1}{r_j}} \tag{4.21}$$

But the packet losses lower the actual throughput of the link. p_l^f and p_l^r are supposed to be the packet success probabilities of link l in the forward and reverse directions respectively, then the ETP of link k is given by:

$$ETP_l = \frac{p_l^{(f)} \times p_l^{(r)}}{b_l} \tag{4.22}$$

In the form of ETX, we have:

$$ETP_l = \frac{1}{ETX_l \times b_l} \tag{4.23}$$

i. e., it is computing the expected throughput of a link directly. $f(P)$, is the throughput of the bottleneck link of the path,

$$f(P) = \min_{k \in P} ETP(K) \tag{4.24}$$

Unlike ETX, ETT and EDR, ETP has a more accurate model for the impact of contention in 802.11 MAC.

4.2.10 Multi-channel Routing Protocol (MCR)

WCETT lacks switching cost so [55] added it in (4.10) and suggested MCR as:

$$MCR = (1 - \beta) \times \sum_{l=1}^{n} (ETT_l + Sc(c_l)) + \beta \times \max_{1 \leq j \leq c} X_j \tag{4.25}$$

The additional component, Switching Cost, $SC(c_l)$ is defined as follows:

$$SC(c_j) = \sum_{\forall i \neq j}^{n} InterfaceUsage(i) \times Switchingdelay \tag{4.26}$$

This value does not figure in the time interval that this interface is tuned to channel j, but is idle. $SwitchingDelay$ is latency for switching an interface and can be measured offline. When a packet arrives on channel j, $\sum_{\forall i \neq j}^{n} InterfaceUsage(i)$ measures the probability that the switchable interface will be on a different channel ($i \neq j$).

Like WCETT, MCR fails to figure the inter-flow interference besides the assumption that all available channels are orthogonal but channel-switching cost makes MCR to be incorporated with the routing protocol like DSR, AODV for multi-channel and channel-switchable wireless network.

4.2.11 Medium Time Metric (MTM)

MTM [56] minimizes the time of consumption of physical medium. Due to the shared nature of wireless networks, not only individual links may interfere (intra-flow interference) but transmissions compete for the medium with each other in the same geographical domain. The longer the physical distance of a hop results in the higher energy consumption and the more other hops are affected. The MTM of a packet p on a path P is defined as follows:

$$MTM(P,p) = \sum_{\forall l \in P} \tau(l,P) \tag{4.27}$$

Where $\tau(l,P)$ is the time required to transfer packet p over link l. $\tau(l,P)$ is defined as:

$$\tau(l,P) = \frac{overhead(l) + \frac{size(p)}{rate(l)}}{reliability(l)} \tag{4.28}$$

Utilizing (4.2) and (4.7), eq.(4.28) in the form of ETX is:

$$\tau(l,P) = ETX(overhead(l) + t) \tag{4.29}$$

Link overhead can be computed from standards and specifications as well as from the type and configuration of the used wireless device. The packet size should be easily available through the routing protocol. Link transfer rate and reliability usually are known to the MAC layer. However, this information often is not accessible to higher network layers because the techniques used for auto-rate selection on the MAC layer are considered proprietary. It is possible to estimate the values for transfer rate and link reliability by probing. Though, this information produces unnecessary overhead and less accurate results than inter-layer communication would.

Therefore, Awerbuch *et al.* [56] would favor that radio card manufacturers provide a standard interface in order to enable access to this information by higher network layers. Although we agree with them principally, one should not expect that all problems of measuring transfer rate or link reliability be solved at once thereby. Awerbuch *et al.* [56] measured an end-to-end throughput which was equal to minimum hop count and ETX in short distances. When the distances were larger, minimum hop count and ETX found routes with a few hops. MTM selected multi-hop paths with more hops but higher capacity. For this reason, the resulting end-to-end throughput was up to 20 times higher with MTM than with the other metrics.

Table 4.1: Performances over ETX, design goals and platforms of the ETX based metrics

Metric	Performance over ETX	Target Platform(s)	Design Requirement(s)
ETX		Multi-Hop Networks	Maximizing throughput
Modified ETX	1.Accurate loss estimation. 2. 50% less packet loss	Time-Varying WMN's	Selecting good paths among time-varying binary symmetric channel
ENT	1.Accuracy of loss estimation 2.50% less packet loss 3.Least link layer trans.	Time-Varying WMNs	Selecting good paths among time-varying binary symmetric channel
ETT	Ignores intra-flow interference	Multi-Radio, Multi-Hop WMNs	Measuring loss rate bandwidth
WCETT	Multiple Channels	Multi-Radio, Multi-Hop WMNs	Choosing high throughput paths
MIC	It Captures both inter-flow and intra- flow interference	Multi-hop Wireless Networks	Efficient calculation for min. weight path + loop-free rtng
iAWARE	Considers intra-flow + inter-flow interference, medium instability, and data-transmission time	Multi-Radio WMNs	Finding better paths with less interflow and intra-flow interference
DBETX	1.Improves performance for varying channels 2.Outperforms ETX for fading.	Fading Wireless Channels	Maximizes throughput for fading Channels
EETT	Select multi-channel routes with least interference to maximize end-to-end throughput.	Large-Scale Multi-Radio Mesh Networks	Maximizing end-to-end throughput
EDR	1.Finds degree of contention 2.Quantifies impact of link loss 3.Considers concurrent trans if links do not interfere	Multi-Hop Wireless Networks	Finding high-throughput paths
ETP	More Accurate Throughput Estimations.	Multi-rate Multi-channel Mesh Networks	Maximizing throughput
MCR	Uses multiple channels using multiple interfaces	Multi-Channel Multi-Interface Ad-hoc Networks	Increase network capacity by multiple channels
MTM	1.MAC-related overhead 2.Gives higher throughput, 3. Selects more reliable links	Multi-rate Ad-hoc Wireless Networks	avoiding long range links selected by shortest path
EstdTT (SrcRR Algorithm)	1.Predicts best 802.11 trans. rate 2.Reduces loss rate avoiding TCPs timeouts and idle time 3. improves the choice of link transmission bit-rate.	802.11 Mesh Networks	Improving varying link loss rates, transient bursts of losses, poor transmit bit-rate selection, failure to identify high throughput routes,
ETX Distance	1.Greedy forwarding based on ETX Distance,2.Outperforms previous geographic routing approaches	Geographical WSN's	Greedy forwarding
Multicast ETX	Improves throughput at cost of min. energy consumption.	Multi-hop Wireless Networks	Energy-efficient reliable comm. for unreliable or lossy link

4.2.12 Estimated Transmission Time (EstdTT)

Neglecting the overhead, Aguayo, Bicket *et al.* [57] assumed 1500 Bytes as a constant size of the packet and suggested Estimated Transmission Time metric is defined as:

$$EstdTT = \frac{1}{reliability(l) \times rate(l)} \tag{4.30}$$

This can alternatively be written using (4.2), as:

$$EstdTT = \frac{ETX(l)}{rate(l)} \tag{4.31}$$

4.2.13 ETX Distance

ETX metric is combined with greedy forwarding to optimize routing path without relying the frequently broadcast route probing messages (as in original ETX) in [58]. ETX virtual distance between pairwise nodes x_i and x_j as the minimal ETX among all the routing paths connecting x_i and x_j, i.e.,

$$\delta(x_i, x_j) = \min_{l_i \in L}(l_i) \tag{4.32}$$

Where L is the set of hops or paths connecting x_i and x_j. It has been suggested that ETX distances between pairwise nodes in a WSN can be inferred from their virtual coordinates. Making the comparison of the ETX distances between neighboring nodes, the greedy forwarding can determine the next hop. ETX distance comparison based greedy forwarding guides a packet towards the correct direction and deliver the packet through consecutive hop by hop forwarding, as ETX distance directly reflects the length of a communication path between pairwise nodes in a WSN.

4.2.14 Multicast ETX (METX)

This energy-efficient routing metric [59], aims to minimize the total transmission energy, in the presence of an unreliable link layer, for the path:

$$C(s,d) = \frac{C(s,u) + W(u,d)}{1 - P_{el}} \tag{4.33}$$

$C(s,d)$is the expected energy-cost of transmission from a source s to destination d, l is the link between u and d in the path, p_{el} is the error rate of that link, and $W(u,d)$ is the transmission energy required between nodes u and d. The authors in [60] modified the metric given by Eq. (4.33) to a new metric, METX setting $W(u,d)$ to 1, as WMNs are not energy sensitive. Eq. (4.34) gives us the total expected number of transmissions needed by all the nodes along a path from a source to a destination in order to guarantee successful reception of at least one packet at the receiver:

$$METX = \sum_{i=1}^{n} \frac{1}{\prod_{j=1}^{n} (1 - P_{ej})} \tag{4.34}$$

In terms of ETX using Eq (4.1):

$$METX = \sum_{i=1}^{n} \frac{1}{\prod_{j=1}^{n} \frac{1}{ETX_j}} \tag{4.35}$$

i denotes the i^{th} link from a source to a destination comprising n links.

4.3 Conclusion

After minimum hop count which usually selects lossy links, ETX is the most widely used routing link metric (in the presence of least mobility of nodes and availability of links). We therefore, analyzed and compared the performance of those wireless routing which are ETX based and are used by the recent routing protocols. Overheads occurred and throughputs achieved due to the factors added to ETX have been listed and discussed. Future work goals are to simulate these metrics with the most widely used protocols, as DSR, AODV, OLSR, etc and to analyze their performance over ETX and minimum hop count.

Chapter 5

Design Requirements for Quality Link Metrics

Contents

5.1 Introduction

Wireless multi-hop networks provide many conveniences to the mobile users, as they can communicate any time without any disruption while moving around. Such networks have some distinguished characteristics as 'dynamic topology', that is the outcome of mobility of random number of nodes, at random times, in random directions, for random periods of time. So, the underlying network demands the routing protocol to dynamically cope with the topological changes. The mobile nodes are very often limited in resources such as CPU capacity, storage capacity, battery power, and bandwidth. This implies that the routing protocol must be able to minimize the control traffic, (as trigger/periodic update messages), delays (due to retransmissions, or computation of metrics), and so on.

The performance of a wireless network depends upon the efficiency of the routing protocol operating it and the most important component of a routing protocol is 'routing link metric'. Because, a link metric first considers the quality routes then decides the best end-to-end path. The link metric plays a key role to achieve the desired performance of the underlying network by making the routing protocol: fast enough to adopt topological changes, light-weight to minimally use the resources of nodes, intelligent to select the fastest path from source to destination among the available paths and capable to enable the nodes to have a comprehensive idea about the topology.

Considering the demands of a wireless multi-hop network from its operating protocol and the factors influencing its performance, the link metric is supposed to fulfill certain requirements. An efficiently designed link metric can better help a routing protocol to achieve appreciable performance from the underlying network by dealing with these issues. In this work, we, therefore, identify the characteristics that must be taken into account while designing a routing link metric. It is worth stating that it is impossible to implement all mentioned characteristics in a single metric. Rather they provide guidelines that might be helpful to design a link metric. By simulation results we have demonstrated that the computational overhead produced by a routing metric may degrade the performance of the protocol. The issues that influence a wireless network, if efficiently tackled, they become the characteristics of the newly developed protocol.

5.2 Related Work and Motivation

After analyzing reactive and proactive protocols, Yang *et al.* [49] proposed that the proactive protocols that implement the hop-by-hop routing technique, as Destination-Sequenced Distance Vector (DSDV) [23] and Optimized Link State Routing (OLSR) [26] protocols are the best choice for mesh networks. They have also inspected the design requirements for routing link metrics for the mesh networks and related them to the routing techniques and routing protocols. In the chapter, four design requirements for link metrics; stability, minimum hop count, polynomial complexity of routing algorithm and loop-freeness have been suggested. However, the focus has only been on the mesh networks. Secondly, all the work is merely restricted to these four requirements. There are several other requirements that may help to achieve global optimization. For example, 'computational overhead' that might be outcome of the mathematical complexity introduced in the link metric or an attempt to design a multi-dimensional metric to tackle multiple issues simultaneously.

Das *et al.* in [61], have discussed the dynamics of the well known metrics: Expected Transmission Count (ETX) [42], Expected Transmission Time (ETT) [47] and Link Bandwidth [43], in real test beds. Across various hardware platforms and changing network environments, they tested two requirements: stability and sensitivity for some existing routing link metrics. Authors have also discussed the dynamics of the above mentioned metrics and tested their performance on the test beds for the above stated requirements. Anyhow, both the design issues of the link metrics and their design requirements are yet to be analysed.

In [62], Yaling *et al.* systematically analyzed the impact of working of wireless routing link metrics on the performance of routing protocols. They related the characteristics of routing metrics to reactive and proactive protocols. They have presented the ways by which the mathematical properties of the weights given to the paths affect the performance of routing protocols. They proposed and discussed three operational requirements: optimality, consistency and loop-freeness. However, these properties do not cover all design requirements; for example, computational overhead, a metric can produce and the performance trade-offs a metric has to make among different network performance factors. For example, a routing protocol achieves higher throughput values at the cost of end-to-end delay or routing overhead. So, instead of generalizing the design requirements, we have pointed-out and analyzed almost all possible design requirements.

5.3 Factors Influencing WMhNs

The factors affecting the wireless networks help to have an idea about the problems they have to face. Along with other protocols that operate a network, routing protocols play a significant role in the performance of wireless multi-hop networks. So, in this section, we state and discuss some general issues regarding wireless networks that will provide a ground to discuss the requirements for designing a routing metric.

(A) In wireless networks, generally the link quality considerably varies in different periods of time. The reasons may be: some mobile nodes are moving randomly, some go-out of range, some intentionally cut-off the ongoing communication, some die-out due to battery and so on. The respective routing protocols must be able to dynamically cop with the situation.

(B) Usually, the behavior of channels varies in links and then in complete paths from source to destination. In the case of Quality of Service (QoS) routing, the the link creating bottle neck for performance must be given attention. Similarly, change in the quality of one link affects the others, as in the case of intra-flow and inter-flow interferences but not in the case of (minimum) hop count.

(C) Upper layer protocols are affected by the choice of a particular link metric at the lower layers [61].

(D) The selection for a particular flow on a particular channel is not random in the case of multiple flows on multiple channels.

(E) The wireless multi-hop networks in which each node is equipped with a single radio interface and all radio interfaces operate on the same frequency channel, often suffer low channel utilization and poor system throughput.

After discussing the behavior of wireless networks, it would be appreciable to discuss and analyze the design requirements for routing link metrics.

5.4 Design Requirements for Routing Metrics

Heretofore, several routing protocols either have been designed from scratch or optimized to improve the performance of a particular wireless network. A routing protocol is responsible to choose the best paths from source to destination. This decision is based upon the information provided by link metric. Therefore, primary emphasis has been given to propose new link metrics of different varieties; a single metric, a single mixed metric, a single compound metric, multiple metrics and a composite metric are few examples that have been designed and implemented with the existing protocols [78]. Thus, while designing a link metric for a routing protocol, following design requirements must be taken into account.

5.4.1 Minimizing hop-count or path length

This is first of the several canonical design requirements, a link metric is supposed to fulfill that has a goal to route packets through minimum weight paths. Often a longer path increases the end-to-end delay and reduces the throughput of a path. So, the respective metric must prefer a path with minimum length over it. This design requirement is implicitly or explicitly attempted by almost all of the existing link metrics. For instance, *ETX* achieves maximum throughput by minimizing the number of transmissions and thus raises a network throughput. Minimum Loss *(ML)* [77] selects the paths with minimum loss rates or higher probabilities of successful transmissions. Now, if all links in some end-to-end paths have the same probabilities of success, then qualities of the paths becomes dependent on the number of hops. *ML* has been implemented with OLSR that prefers minimum hop path in this case. Hop count is the most widely used metric in MANET routing protocols [28], as all of the RFC's prefer to use hop count as a routing metric for the sake of simplicity and least computational overhead.

5.4.2 Balancing traffic load

To achieve appreciable throughput, the respective metric can be designed to ensure that no node or link is disproportionately used by minimizing the difference between the maximum and minimum traffic load over the nodes or links.

When a link becomes over-utilized and causes congestion, the link metric can choose to divert the traffic from the congested path or overloaded nodes to the underloaded or idle ones to ease the burden. Table.5.1 lists some examples of the load balancing routing protocols.

Table 5.1: LOAD BALANCING PROTOCOLS WITH RESPECTIVE TECHNIQUES

Routing protocol	**Load balancing Techniques**
Load Balanced Ad-hoc Routing (LBAR) [63]	Measuring activity of a node participating in the communication
Load Sensitive Routing(LSR) [64]	Counting the total number of packets both at the queue of mobile node and neighboring node
Dynamic Load Aware Routing (DLAR) [65]	Measuring the routing over head at the intermediate node
Simple Load-balancing Ad-hoc Routing (SLAR) [66]	Measuring the forwarding load of the mobile nodes
Ad-hoc On-demand Distance Vector Routing with Load Balancing (LB-AODV) [67]	Analyzing the balance index
Load Aware Routing in Ad-hoc networks (LARA) [68]	- Measuring hop counts and traffic loads for TCP source - Measuring level of contention for non TCP sources
Simple Load-balancing Approach (SLA) [69]	Measuring the traffic load at the mobile node
Delay-based Load-Aware On-demand Routing (D-LAOR) protocol [70]	Measuring the hop count and end-to-end delay

5.4.3 Minimizing delay

A network path is preferred over the others because of its minimum delay. It is worth noting that if intra-flow and inter-flow interferences, queuing delays, and link capacity are not taken into consideration, then delay minimization often ends up being equivalent to path length or hop-count minimization.

5.4.4 Maximizing data delivery/aggregating bandwidth

Maximizing the probability of data delivery, minimizing the probability of data loss, minimizing the packet loss ratio, maximizing the packet delivery fraction, maximizing the individual path throughput, increasing the network capacity, are the same and utmost important features, a wireless routing protocol is expected to implement. So, in wireless networks, the attempt has always been to choose an end-to-end high capacity path. A protocol can achieve maximum throughput:

(a) directly by maximizing the data flows,

(b) indirectly by minimizing interference or retransmissions,

(c) allowing the multiple rates to coexist in a network, where a higher channel rate is used over each link. It is possible if more packets can be delivered in the same period with the consideration of packet loss rates [71], data can be splitted to the same destination into multiple streams, each routed through a different path. End-to-end delay may also be reduced as a direct result of larger bandwidth.

5.4.5 Minimizing energy consumption

Energy consumption is a major issue in all types of wireless networks where the battery lifetime constrains the autonomy of network nodes. A protocol, if chooses path with an unreliable link, it would probably produce longer delay due to higher retransmission rates, that ultimately results in raise in energy consumption (along with computational processing overhead of aggressive control packets). For energy saving, most of the work focuses on the communication protocol design. For example, the routing protocol ZigBee [72] uses a modified AODV to be used by low-power devices. By adapting transmission power to the workload, Real-time Power-Aware Routing (RPAR) protocol [73] reduces communications delays.

5.4.6 Minimizing channel/interface switching

Both in single-hop and multi-hop wireless networks, for the maximum utilization of available bandwidth, one way is to use as many channels as possible depending upon the sophistication of the technology. In this case, the different data flows are to be switched on different channels, resulting in some delay. So, the phenomenon may be given attention by the respective metric.

When using multiple channels, two adjacent nodes can communicate with each other only if they have at least one interface on a common channel. So, it may be necessary to periodically switch interfaces from one channel to another with the production of a delay. In [55], Vaidya *et al.* used an interface assignment strategy that keeps one interface fixed on a specific channel, while other interfaces can be switched among the remaining channels, when necessary.

5.4.7 Minimizing interference

Bandwidth of a wireless link is shared among neighboring nodes, so, the contending nodes have to suffer from the inter-flow interference. The channels on the same link are always being disturbed from the intra-flow interference. Both intra-flow and inter-flow interferences may result in bandwidth starvation for some nodes as they may always find the available channels busy. Hence, both of the diversity of channel assignments and the link capacity possibly need to be captured by the link metric, as Yang *et al.* have presented in their work [48].

5.4.8 Minimizing the Computational overhead

While designing a routing metric, necessary computations should be considered that must not consume memory, processing capability and the most important; battery power. For example, we discuss the case of three widely used routing link metrics for wireless routing protocols: *ETX*, its inverse, say, *invETX* and *ML*.

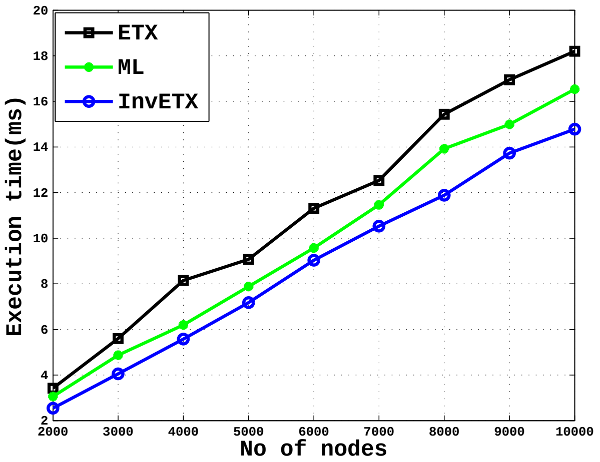


Figure 5.1: Computational overhead generated by ETX, ML and InvETX

For an end-to-end path, P_{e2e}, these metrics are expressed by the following equations:

$$ETX_{P_{e2e}} = \sum_{l \in P_{e2e}} \frac{1}{(d_f^{(l)} \times d_r^{(l)})} \tag{5.1}$$

$$invETX_{P_{e2e}} = \sum_{l \in P_{e2e}} (d_f^{(l)} \times d_r^{(l)}) \tag{5.2}$$

$$ML_{P_{e2e}} = \prod_{l \in P_{e2e}} (d_f^{(l)} \times d_r^{(l)}) \tag{5.3}$$

Where $(d_f^{(l)} \times d_r^{(l)})$ is the probability of success for delivery of probe packets (134 *bytes* each) on the link l on P_{e2e} from source to destination (forward direction) and from destination to source (reverse direction).

Regarding the computational complexity, all of the three metrics have to calculate the equal number of products $(d_f^{(l)} \times d_r^{(l)})$ for the same number of links. But ETX has to suffer from more computational overhead (inverse and sum of n products) than ML (multiplication of n products only). Similarly, ML generates more computational overhead than $invETX$. As a result, $invETX$ achieves higher throughputs than ML and ETX. Similarly, ML performs better than ETX. The computational overheads generated by the three metrics have been shown in Fig.5.1. Along with other implementation parameters, the amount of computational load generated by each metric influences its performance accordingly. This fact can be seen in Fig.5.2, 5.3 and 5.4. This overhead is directly proportional to the number of nodes/links.

5.4.9 Maximizing route stability

Unlike wired networks, frequent topological changes in the wireless links may not only huge generate routing load but may also slow down the convergence of the respective routing protocol operating the network. The stability of the paths is found by the path characteristics that are captured by the routing metric that can be either load sensitive or topology-dependent [49]. Former type of metrics assign a weight to a route according to the traffic load on the route. This weight may change frequently as the link break and establish. On the other hand, topology dependent metrics assign a weight to a path based on the topological properties of the path, such as the hop-count and link capacity of the path. Therefore, topology-dependent metrics are generally more stable, especially for static networks where the topology does not change frequently. Load-sensitive and topology-dependent metrics are best used with different types of routing protocols, since routing protocols have different levels of tolerance of path weight instability [74].

5.4.10 Maximizing fault tolerance/minimizing route sensitivity

In the case of multi-path routing, the link metric can provide fault tolerance by having redundant information of the alternative paths. This reduces the probability that communication is disrupted in the case of link failure. To reduce the network load due to the redundancy, source coding can be employed with the aid of some sophisticated algorithms with compromising on the issue of reliability. Such type of raise in route resilience usually dependents upon the diversity, or disjointness like metrics for the available paths [74].

5.4.11 Avoiding short and long lived loops

A metric can better help a routing algorithm to avoid forwarding loop (both short lived and long lived) to minimize the packet loss. Because selecting redundant links degrades the performance of the network due to more path lengths and consequently increased end-to-end delay. For example, Fahim *et al.* [75] have addressed the problem of transient mini-loop problem that takes place because of fisheye scoping in Fish Eye OLSR (OFLSR) protocol. They have provided a potential solution that enables the routers to calculate ”safe” scope for a particular topology for all updates. The minimum TTL value that eliminates mini-loops, is calculated in distributed fashion by all mesh routers in advance at the ”scope” boundary. Independent of the scale of network, keeping efficiency of the algorithm as before, the authors improved the safety of OFLSR.

5.4.12 Considering performance trade-offs

Generally, a protocol achieves higher throughput values at the cost of increased end-to-end delay in the case of static networks. Whereas, in mobile networks, the frequent link beaks cause more routing overhead to obtain better throughput from the network. To discuss such type of trade-offs, we have set-up a simulation scenario that is discussed in the following subsection:

Simulations

We use the implementation of *ETX*, Minimum Delay *(MD)* [76], and *ML* [77] with OLSR [78] in NS2-2.34. Then we implement the fourth metric, *invETX*, as expressed by eq. (5.2). In the area of $1000m$ x $1000m$, 50 nodes are placed randomly to form a static network. Constant Bit Rate (CBR) traffic is randomly generated by 20 source-destination pairs with packet size of $64bytes$. Each simulation is performed for five different topologies for $900s$ each. Then the average of five different values of each performance parameter is used to plot the graphs. To observe the performance of OLSR with four metrics, we randomly generated the data traffic with number of packets from 1 to 16 per second.

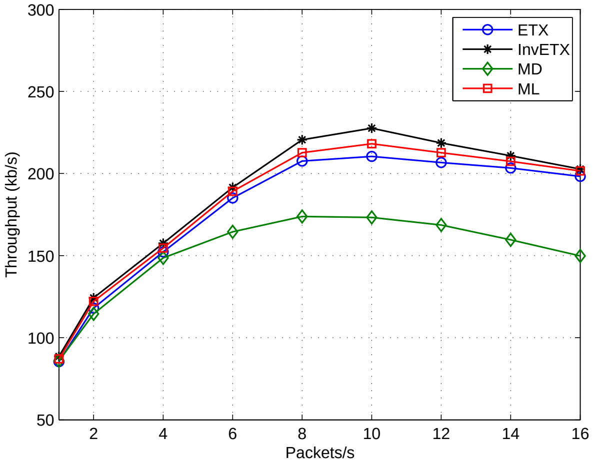


Figure 5.2: Average throughput by OLSR with *ETX*/*invETX*, *MD* and *ML*

To better understand the performance trade-offs, we take an example of the static wireless multi-hop networks that have two major issues; bandwidth and end-to-end delay. In this type of networks, the proactive protocols are preferred due to stability, like, OLSR, instead of the reactive ones that are suitable for the environments where topology changes frequently due to mobility. Moreover hop-by-hop routing technique helps OLSR to handle aggressive overhead as compared to source routing. Using the Multi-point Relays (MPRs) selection along with proactive nature, OLSR achieves minimum delay. In the following subsections, we discuss the performance parameters; throughput, End-to-End Delay *(E2ED)*, and Normalized Routing Load *(NRL)*.

Throughput

In static networks, with varying data traffic rates, *OLSR-MD* produces lowest throughput as compared to *OLSR-ETX/OLSR-invETX* and *OLSR-ML*. Moreover, in medium and high network loads, there are more drop rates as compared to small load in the case of *MD* metric.

This is due to the one-way delays that are used to compute the *MD* routing metric with small probe packets before setting up the routing topology and not considering the traffic characteristics. It may thus happen that, if no other traffic is present in the network, the probes sent on a link experience very small delays, but larger data packets may experience the higher delay or retransmission due to congestion. Thus, *OLSR-MD* is not suitable for the static networks with high traffic load, as, it degrades the network performance by achieving less throughput values. The *OLSR-ML* in medium and high network loads produces higher throughput values because *ML* attains the less drop ratios as compared to *ETX*. Moreover, in *ML* the paths with minimum loss rates or higher probabilities of successful (re)transmissions lead to high data delivery rates, with an additional advantage of more stable end-to-end paths and less drop rates.

E2ED

OLSR-MD uses the Ad-hoc packet technique to measure the one-way delay. Then proactive delay assurance approach is used to measure *MD* metric. The minimum delay metric performs best in terms of average packet loss probability. In Fig. 5.3, *OLSR-MD*'s delay is showing the lowest values among other metrics. This is due to the route selection decision based on delay of ad-hoc probes. While *OLSR-ETX* and *OLSR-ML* produce increasing value of delay, when traffic increases. The very first reason is that both metrics have no mechanism to calculate the round trip, unlike *MD* metric. Meanwhile, in *ML*, selection of longer routes with high probability of successful transmission augments the delay as compared to *ETX*.

NRL

OLSR-MD suffered from the highest routing loads. As, ad-hoc probes are used to measure the metric values and are sent periodically along with TC and HELLO messages. On the other hand, *OLSR-ETX* and *OLSR-ML* calculate the probabilities for the metric from the values obtained from the enhanced HELLO messages. OLSR uses HELLO and TC messages to calculate the routing table and these messages are sent periodically. The

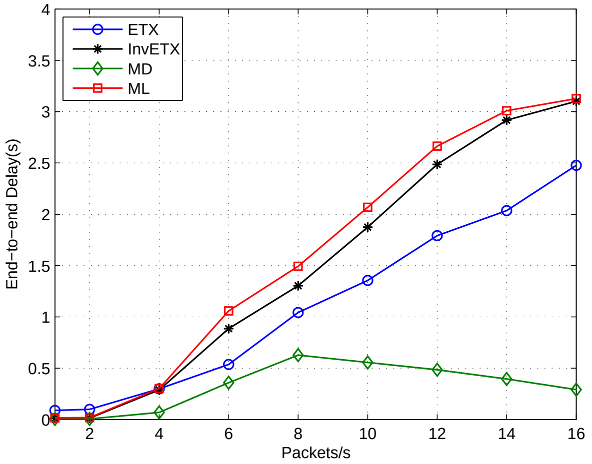


Figure 5.3: Average End-to-end Delay caused by OLSR with the four metrics

delivery ratios are measured using modified OLSR HELLO packets that are sent every t seconds ($t = 2$, by default). Each node calculates the number of HELLO messages received in a w second period ($w = 20$, by default) and divides it by the number of HELLO messages that should have been received in the same period (10s, by default). Each modified HELLO packet notifies the number of HELLO messages received by the neighbor during the last w seconds, in order to allow each neighbor to calculate the reverse delivery ratio. The worse the link quality, the higher the ETX link value. A link is perfect if the ETX value is 1 and its packet delivery fraction is also 1, i.e., no packet loss. On the other hand, if in $wseconds$ period a node has not received any HELLO message then ETX is set to 0 and the link is not considered for routing due to 100% loss ratio. Thus, due to no extra overhead to measure the metric *OLSR-ETX*/*OLSR-invETX* and *OLSR-ML* have to suffer from low routing load as compared to *OLSR-MD*.

The ad-hoc probe packets are sent by *MD* to accurately measure the one-way delay. Thus, low latency is achieved by selecting the path with less Round Trip Time (RTT). On the other hand, these ad-hoc probes cause routing overhead in a network and decrease the throughput when data load is high in a static network.

In static networks, to measure an accurate link with less routing load is a necessary condition. The delay cost due to increase in the number of intermediate hops is paid to achieve throughput by *OLSR-ML*. As *ML* selects those paths which possess less loss rates,

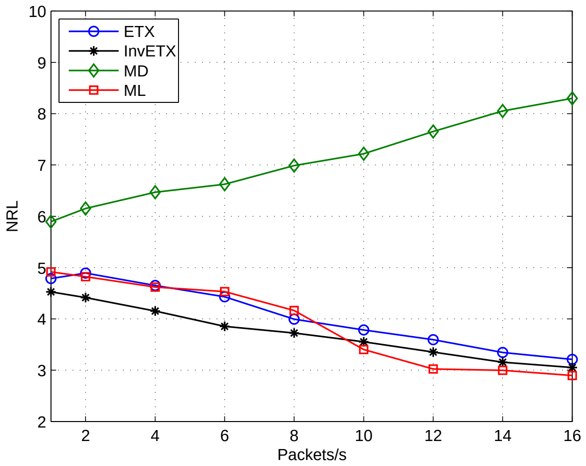


Figure 5.4: Average NRL generated by OLSR with *invETX*, *ETX*, *MD* and *ML*

therefore, a longer path with high successful delivery is preferred. Thus the product of the link probabilities selection decreases the drop rates and increase the RTT.

OLSR-ETX uses the same mechanism to measure the link quality as that of *OLSR-ML*, i.e., modified HELLO messages. But summing up the individual probabilities and preference of the shortest path reduces the delay of *ETX* as compared to *ML*. Thus, a slow link preference results more drop rates of *OLSR-ETX* as compared to *OLSR-ML*.

This sort of trade-off is common in routing protocols. While designing a link metric, if demands of the underlying network are taken into consideration then it becomes easy to decide that among which performance parameters, trade-off(s) should be made. For example, *ML* and *ETX* achieve higher throughput values than *MD*, as shown in Fig.2, whereas *MD* remarkably achieves less end-to-end delay than *ML* and *ETX* that is depicted in Fig.5.3.

In Table 5.2, we provide a list of routing link metrics and routing algorithms that have taken into account some of the design requirements suggested in this chapter.

5.5 Conclusion

Along with the importance of a routing protocol, an issue of an efficiently designed routing link metric runs parallel. In this chapter, we therefore, have presented a comprehensive

Table 5.2: Metrics and Algorithms implementing different Design Requirements

Design Requirement	**Metric/Algorithm**
Minimizing hop count or path length	Hop count [28]
Minimizing delay	Per hop RTT [47]
Minimizing packet loss ratio	Interference clique transmissions [79]
Balancing traffic load	MIC [48]
Maximizing the probability of data delivery	Per hop PktPair [47], ML [77]
Maximizing path capacity	Network characterization with MCMR [80]
Aggregating bandwidth/ maximizing fault tolerance	Multipath routing scheme for wireless ad-hoc networks [77]
Maximizing individual path throughput	ETX [42]
Maximizing individual	ETX [42]
Maximizing network throughput	ETX [42]
Maximizing individual	ETX [42], Per hop RTT [47]
Minimizing interference	iAWARE [50]
Minimizing channel switching	MIC [48], WCETT [47]
Minimizing interface switching	MCR Protocol [55]
Maximizing route stability	Link affinity metric [81]
Minimizing energy consumption	MTPR [82], MBCR [83]
Avoiding routing loops	Loop avoidance for Fish-Eye OLSR in Sparse WMN's [75]
Minimizing computational overhead	ML [77]

study on the design requirements for routing link metrics in a broader view. We discussed several possible issues regarding wireless networks that can better help in designing a link metric. The ambition of a high throughput network can only be achieved by targeting a concrete compatibility of the underlying wireless network, the routing protocol operating it, and routing metric; heart of a routing protocol. Depending upon the most demanding features of the networks, different routing protocols impose different costs of 'message overhead' and 'management complexity'. These costs help to understand that which type of routing protocol is well suitable for which kind of underlying wireless network and then which routing link metric is appropriate for which routing protocol. In future, we are interested in an analytical study of such kind of compatibility.

Chapter 6

Interference and bandwidth adjusted ETX (IBETX)

Contents

6.1 Introduction

Wireless multi-hop networks consist of wireless nodes that are not in the transmission range of each other. So, the intermediate nodes act as routers to receive and send the routing and data packets from and to the nodes in their transmission range. In order to have appreciable performance from the underlying wireless network, the routing protocol that is responsible to operate it, plays a key role. The heart of a routing protocol is the link metric. The Minimum Hop-count is the most popular and IETF standard metric [28] and is appropriately used by Wireless Ad-hoc Networks, where the objective is to find new paths as rapidly as possible in the situations where quality paths could not be found in due time due to higher rates of node mobility. Secondly, hop-count is simple to calculate and it avoids any computational burden on the routing protocol. But in the case of Wireless Mesh Networks (WMNs) mobility is not an issue; where either stationary or minimally mobile nodes interconnect and form a wireless backbone. Now, depending upon the demands of a static wireless multi-hop network; low end-to-end delay and high throughput, the routing protocol must choose a realistic routing link metric to select the quality links. Several newly proposed metrics [84], [52], [54] have succeeded to find the quality paths more efficiently than the previous ones [42], [45], [47]. Since, we are dealing with the static wireless networks where all nodes are broadcasting by nature and the links do not have the same characteristics, therefore, the nodes have to compete for the transmission opportunities with their neighbors resulting in contention. Consequently, the network performance degrades mainly because of two issues; firstly, the links with lower bit rates lower the performance of faster links, secondly, the interference causes congestion and collisions that pretend the medium to be busy. Heretofore, none of the work has considered both of the phenomena simultaneously.

In this chapter, we propose a new routing metric, interference and bandwidth adjusted ETX (IBETX), that selects the optimal paths in the wireless multi-hop networks. As longer paths usually achieve higher throughputs, the metric takes them into consideration while selecting the best path, unlike Expected Transmission Count (ETX) [42] (and all ETX-based metrics that do not explicitly handle interference and are unable to consider longer paths). Like [84], our metric is hybrid; it is load-dependent and takes care of link quality as well. The routing layer can give appreciable performance in multi-hop networks, if it takes the relevant information from the MAC layer. For IBETX to have more accurate information, we used cross-layer to take the wireless link information from MAC layer. Then we use this information to compare the links by their transmission rates and the amount of contention they have, by measuring the interference.

6.2 Related Work

In recent years, though many quality link metrics have been proposed, still minimum hop-count is widely used by existing wireless routing protocols. Using this metric, the source node selects the least hop route to the destination node. The metric blindly selects minimum hop routes without comparing the loss ratios of the competing routes. This may increase the number of retransmissions causing loss in throughput and resulting degradation in the overall performance of the underlying network. To overcome this problem, ETX metric is proposed by De Couto *et al.* [42]. It is the expected number of (re)transmissions required to successfully transfer a packet over a link. The link interference is not taken into consideration by the ETX metric. The authors in [84] proposed ELP to find optimal paths in a mesh network. To estimate link performance, ELP uses both link traffic and link quality information. It does not consider the bandwidth of the contending links. Draves *et al.* [47], proposed Expected Transmission Time (ETT) that is multiplication of ETX with the link bandwidth to obtain the expected link airtime for the successful transmission of a packet. The interference is not taken into account by ETT, thus, another metric Weighted Cumulative ETT (WCETT) [47] tackles this issue along with using ETT. One of the limitations of ETX is that it may not follow the link quality variations. So, Modified ETX (mETX) and Effective Number of Transmissions (ENT) have been proposed in [45] that are aware of the probe size. These metrics consider the standard deviation to project physical-layer variations along with the link-quality average values. But inter-flow interference handling mechanism is not present in WCETT. The authors in [48] and [49] proposed the Metric of Interference and Channel-switching (MIC). It tackles the issue of inter-flow interference and guarantees the shortest paths by calculating the interference due to the neighbors and selects the minimum-cost paths by the help of MIC virtual nodes. mETX and ENT metrics do not take into account the intra-flow interferences, therefore, Interference AWARE (iAWARE) [50] estimates the average time for which the medium remains busy because of (re)transmissions from each interfering neighbor. To measure the effects of variations in the routing metrics due to continuously produced interference by neighboring nodes, this metric uses Signal to Noise Ratio (SNR) and Signal to Interference and Noise Ratio (SINR).

While counting the number of (re)transmissions required to transmit a data packet, ETX does not consider the maximum number of MAC-layer retransmissions. Therefore, Distribution Based Expected Transmission Count (DBETX) [51] performing the cross-layer optimization, achieves higher network throughputs in the presence of fading while channels are continuously changing their behavior. ETT is not able to evaluate multi-channel paths precisely when the paths are long. The authors in [52] proposed Exclusive Expected Transmission Time (EETT) to select multi-channel routes with the least interference when channels are distributed on a longer path to maximize the end-to-end throughput. Therefore, this metric takes into consideration the channel distribution on long paths that are critical in Large Scale Multi-radio Mesh Networks (LSMRMNs). But DBETX and EETT can not consider the longer paths due to not implementing any mechanism to calculate the interference among wireless neighboring links. ETX is not designed to consider the multi-rate links, so, Expected Data Rate (EDR) [53] took Transmission Contention Degree (TCD) into account. This metric is used for making conservative estimates for paths longer than 3-4 hops by combining time-sharing effects of MAC like Medium Time Metric (MTM) [55] that also minimizes the consumption time of the physical medium by avoiding longer paths.

ETX performs poor transmission bit-rate selection at the 802.11 level. Therefore, Estimated Transmission Time (EstdTT) [57] assumed the size of the packet to be constant of $1500 bytes$ by neglecting the overhead. ETX is designed for single radio, single channel environment. For better utilizing the bandwidth in the case of multiple channels, interface switching is required and then cost of interface switching is to be considered. Multi-channel Routing Protocol (MCR) [55] takes into consideration the interface switching cost and selects channel diverse routes. To improve routing path without relying the frequently broadcast route probing messages (as in original ETX), the ETX metric is combined with greedy forwarding (ETX Distance metric) in [58]. But the metric makes no calculation to measure the bandwidth of the contending links and nodes. In [59], ETX is optimized for energy-conservative networks and named as Multicast ETX (METX). It is an energy-efficient routing metric and reduces the total transmission energy in the existence of an unreliable link layer. The bandwidth sharing of 802.11 is not taken into account by the ETX, so, Expected Throughput (ETP) is proposed by Vivek P. *et al.* [54]. It is a MAC-aware routing metric. This metric takes into consideration the nominal bit rates of the contending links in the neighborhood of a given link. But like ETX and ETT, it also does not consider interference. Table.1. lists the existing metrics along with the issues they have not considered.

Table 6.1: Shortcomings in ETX-based Metrics

Issue(s) not considered	**Metric**
Inter-flow interference	ETT [47] DBETX [51] EETT [52] EstdTT [57] MCR [55] METX [59] ETP [54]
Bandwidth	ELP [84] WCETT [47] MIC [48], [49] iAWARE [50] EDR [53] ETX Dist [58]
Link asymmetry	MTM [56]
Bandwidth and Inter-flow interference	ETX [42] mETX and ENT [45]

6.3 Motivation

This section states and discusses the weaknesses in the existing metrics that are the reasons to propose IBETX. The working principle behind the minimum hop-count implicitly states that whether a path works well or it doesn't work at all, it is selected among a set of available paths based on the least number of hops. Being a non-quality link metric, it does not compare the transmission rates, packet loss ratios and interference due to neighbors on different links. Maximum network performance can be achieved by the respective routing protocol operating the underlying network. The routing protocol performs efficient routing provided that the link metric implemented with it can efficiently find quality paths. ETX augments the throughput of multi-hop paths two times as that of minimum hop-count metric by selecting the quality links [42]. ETX and ETX-based metrics [28] have to face many issues but we only discuss those deficiencies that are once overcome, will improve the metric efficiency and consequently performance of the network. These weaknesses are listed below.

(A) ETX sums the (re)transmission counts of all the links to find the transmission count of the entire path by assuming that all the links on that particular path contend with each other. This is true for less hop paths but is not applicable for longer paths because longer paths have more links that are not in the same contention domain [54]. This spatial reuse implies that the actual transmission cost of a path is less than the sum of the transmission counts of all the links of the path. Thus, adding the ETX of all the links of a path unfairly increases the cost of longer paths due to more packet drops. In other words, ETX penalizes routes with more hops [42]. So, the metric does not consider the longer paths to select the best one. This deficiency of ETX and ETT has been depicted in Fig.1. In the figure, there are three available paths from source to destination. ETX and all those ETX-based metrics that do not take inter-flow interference into account, would select one of the paths between $Path1$ and $Path3$ and would penalize $Path2$. It is obvious from the figure that $Path2$ has multiple contention domains (CDs). The transmissions on a link in $CD1$ do not interfere the transmissions taking place on a link in $CD3$. As a whole, $Path2$ has interference value comparable to that of $Path1$ and $Path3$ or even less. As longer paths have higher throughput [42], [56] but are ignored by ETX, so, $Path2$ is never selected for data transmissions.

(B) ETX, ETT and ETP do not explicitly implement any mechanism to encounter interference that usually becomes performance bottleneck in the wireless static networks.

(C) ETX and ETT do not take any information from the MAC-layer that makes the computations more robust at the routing layer.

(D) ETX and ELP are not capable of differentiating among the transmissions taking place on the links in the same contention domain. Being unable to calculate the bandwidth of the contending nodes, ETX and ELP do not consider the longer paths. Though, the later one takes into account the longer paths by implementing the interference but the former one still remains unable to take the longer paths into account. So, ETP tackles this issue and takes the bandwidth values of the contending links into account. The model proposed in ETP [54] considers the reduction in successful data delivery due to contention from

the slow links and expects the better routes than ETX and ETT. An obvious problem of ETP, like ETX and ETT is that it does not take interference into consideration. Usman *et al.* [84] keeping this issue in view, proposed a new metric, ELP, that calculates interference among the wireless nodes in the same contention domain. But ELP does not provide any mechanism to take transmission rates of the contending links into account. Secondly, it increases computational burden in the algorithm by generating probes of different sizes. Thirdly, the way by which it tunes the delivery ratios (keeping $\alpha = 0.75$), is useful in congested networks only, that is not always the situation.

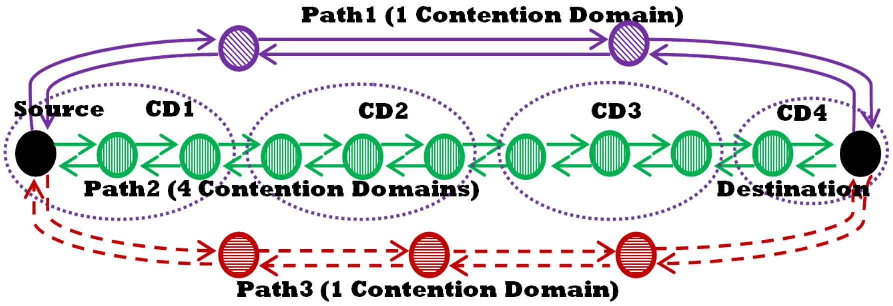


Figure 6.1: Shortcoming of ETX and ETX-based metrics to ignore longer paths

In the next section we discuss our proposed metric that along with measuring delivery ratios, incorporates the two-fold MAC-layer interaction to calculate bandwidth and interference among the contending nodes.

6.4 Interference and Bandwidth Adjusted ETX (IBETX) Metric

We understand that finding the delivery ratios is the primary quantity of interest for selecting quality links. Then comes the issue of contention due to neighbors in a wireless medium. Third most important task is to find high throughput paths that are ignored by ETX. Keeping these concerns in view, IBETX is designed as threefold metric. Firstly, it directly calculates the Expected Link Delivery (ELD), d_{exp}; that avoids the computational burden, as generated by ETX and bypasses the congested regions in the network like ETX. Secondly, it provides the nodes with the information of nominal bit rates and makes them able to compute Expected Link Bandwidth (ELB), b_{exp}, of all the wireless links in the same contention domain by cross layer approach. Thirdly, long-path penalization by ETX is encountered by calculating the interference, I_{exp}, named as Expected Link Interference (ELI) also by cross-layered approach. Then we define IBETX as follows:

$$IBETX = \frac{d_{exp}}{b_{exp}} \times I_{exp} \tag{6.1}$$

Following sub sections give the details that how above given three mechanisms help IBETX to achieve the performance gains.

6.4.1 ELD

This part of the metric finds the paths with the least expected number of (re)transmissions, that may be used onwards for data packet delivery. In other words, the metric estimates the number of required retransmissions calculating the delivery ratios in forward direction by d_f and in reverse direction by d_r of a wireless link mn, as given below:

$$d_{exp}(mn) = d_f \times d_r \tag{6.2}$$

Besides the presence of losses, the main objective of this part is to find the paths with high throughput. To compute d_f and d_r, each node broadcasts a probe packet ($134byte$) every second. Each probe keeps the number of probes previously received from each neighbor in the last $10s$. Thus each node remembers the loss rates of probes on the links to all neighbors in both directions. The quantity d_{exp} in addition to considering lossy links also helps to decrease the energy consumed per packet, avoiding retransmissions. It detects and suitably handles asymmetry by incorporating loss ratios in both directions. It does not route around congested links by avoiding the oscillations that cause more end-to-end delay and by selecting the routes which are either idle or they have less traffic to pass with better delivery ratios by increasing the throughput and better utilizing the network.

This is true that $ETX = \frac{1}{d_f \times d_r}$ produces more overhead than minimum hop-count metric but this overhead is negligible, when compared to the raise in throughput. Keeping this in view, *ELD* not only achieves higher throughput values than hop-count but also over performs ETX. Because, *ELD* avoids the computational overhead generated by ETX that first takes inverse of all d_{exp}'s and then adds them up, whereas, *ELD* only takes their sum. Our network consists of 50 nodes, where this overhead is small but in general, this overhead is directly proportional to the number of nodes or links. This fact is depicted following in Fig. 6.2.

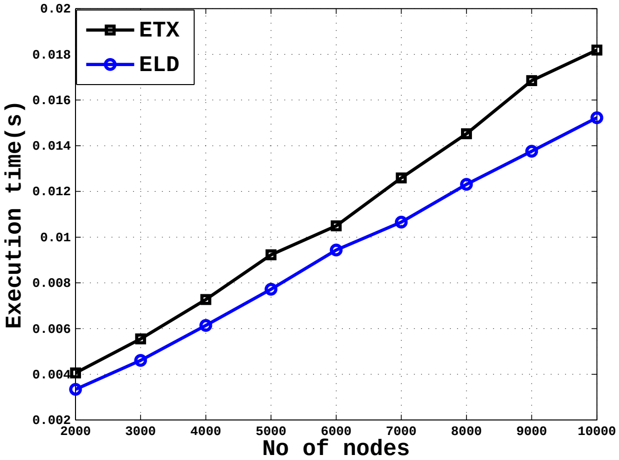


Figure 6.2: Comparison of computational overhead generated by ETX and ELD

6.4.2 ELB

In the wireless environment, slow links lower the bandwidth of the faster ones in their neighborhood. Consequently, all contending links get the same probabilities for transmission due to underlying 802.11 Distribution Coordination Function (DCF) mechanism [54]. This means that nominal bit rate information of the contending links is an important link quality factor. Suppose, we are interested to find the best path between two nodes m and n among a set of contending links either on a source-destination path P or on a non source-destination path NP but in the same contention domain. Then the expected bandwidth of the link mn can be written in the following way:

$$b_{exp}(mn) = \frac{1}{\sum_{i \epsilon P \cap NP} \frac{1}{r_i}} \tag{6.3}$$

Here r_i is the transmission rate of the i^{th} link in the domain $P \cap NP$. Thus capturing the bandwidth sharing mechanism of 802.11 DCF, $b_{exp}(mn)$ considers the accurate throughput reduction of the faster links due the slower ones and predicts the better routes. Moreover, $b_{exp}(mn)$ also encounters the longer paths that are ignored by ETX and ETX-based metrics, as shown in Fig.6.1.

6.4.3 ELI

The delivery ratio $d_{exp}(mn)$ and bandwidth $b_{exp}(mn)$ calculated in the previous subsections help to directly achieve the primary objective, i.e., quality routes but they do not explicitly reveal interference of the links. Interference helps to consider the longer paths ignored by ETX and all those ETX-based metrics that do not calculate the interference among the neighbor links. To exactly measure the congestion in the medium and collisions due to hidden nodes, interference also finds the optimal paths in the wireless network. Moreover, since the probes used to calculate $d_{exp}(mn)$ are very small in size, so, they are successfully received even in a congested network, by depicting the wrong image of link qualities. For example, if a link has only capacity to carry probe packets, it pretends the congested link to be quality link because of its high delivery ratios. Infect, it is not able to carry data packets [84]. We, therefore, incorporate a mechanism to calculate the interference in our metric and define *ELI* that is an expected value calculated by all the nodes on the same source-destination path.

The 802.11's basic Medium Access Control (MAC) is DCF that besides enabling the nodes to sense the link before sending data, also avoids collisions by employing the virtual carrier sensing. DCF achieves this using Request To Send (RTS) and Clear To Send (CTS) control packets that consequently set the Network Allocation Vector (NAV), i.e., $NAV = \tau_{RTS} + \tau_{CTS}$. The NAV is a counter kept that is and maintained by all nodes in the domain with an amount of time that must elapse until the wireless medium becomes idle. Any node can not transmit until NAV becomes zero. It stores the channel reservation information to avoid the hidden terminal problem. Using the cross-layer approach, DCF periodically probes the MAC to find the time period for which the link is busy; τ_{busy}. The interference, a node m has to suffer, is expressed as:

$$i_m = \frac{\tau_{busy}}{\tau_t} \tag{6.4}$$

Where τ_{busy} is the is the duration for which the medium remains busy; in the case of receiving packets it is R_x state (or communication is going-on with other nodes) and the NAV pending. In the interference expression for node m, τ_t is the total window time ($10s$). If a node n is at the transmitting end, its τ_{busy} is given as: $\tau_{R_x} + \tau_{T_x} + \tau_{RTS} + \tau_{CTS}$. Thus the interferences for sending node n and receiving node m are given as:

$$i_m = \frac{\tau_{R_x} + \tau_{RTS} + \tau_{CTS}}{\tau_t} \tag{6.5}$$

and

$$i_n = \frac{\tau_{R_x} + \tau_{T_x} + \tau_{RTS} + \tau_{CTS}}{\tau_t} \tag{6.6}$$

$$i_{mn} = Max(i_m, i_n) \tag{6.7}$$

The link mn formed by nodes m and n are suffering from an interference, i_{mn}, that is the maximum of the interferences calculated in eq.(6.5) and eq.(6.6), is calculated by eq.(6.7).

The receiving node m saves the information of interference computed by eq.(6.5) and sending node n by eq.(6.6). Then we calculate the expected interference of the link mn as:

$$I_{exp}(mn) = \frac{i_{mn}}{1 + i_{mn}} \tag{6.8}$$

Being shared in nature, wireless medium has a problem of interference due to contention. This causes packet loss due to collisions that consequently reduces the bandwidth of links. We, therefore, added I_{exp} factor, that handles the inter-flow interference among the contending nodes. As discussed in section III, the longer paths with higher throughputs are ignored by ETX and ETX-based metrics (as shown in Fig.6.1), ELI would not let any path (independent of number of hop-counts) to be ignored while selecting high throughput paths.

IBETX value for the end-to-end path P is calculated by eq.(6.9), where mn's are the links on P.

$$IBETX(P) = \sum_{mn=1}^{n} IBETX(mn) \tag{6.9}$$

Then the routing metric for the best path P_{best} from source to destination is the minimum value of all available P's. As given below:

$$f(P_{best}) = \min(IBETX(P_1), IBETX(P_2), , , , , IBETX(n)) \tag{6.10}$$

Hence, directly calculating the loss probability, expected bandwidth and expected interference based on the degree of contention present on the links, IBETX successfully finds the quality links.

6.5 Simulations

This section provides the details concerning the simulation environment. We implement and compare the performance of our proposed metric IBETX with ELP, ETX, and ETP in NS-2.34. The window w used for link probe packets is chosen to be of size $10s$ and is named as τ_t, as discussed in the last section. The wireless network consists of 50 nodes randomly placed in an area of $1000m$ x $1000m$. The 20 source-destination pairs are randomly selected to generate Continuous Bit Rate (CBR) traffic with a packet of size $640bytes$. To examine the performance of metrics under different network loads, the traffic rate is varied from 2 to 10 packets per second. For each packet rate, the simulations are run for five different topologies for $900s$ each and then their mean is used to plot the results.

Wireless networks suffer from bandwidth and delay. Because of on-demand nature, the reactive protocols are best suited to cope with these issues for mobile scenario where change in topology is frequent. We are dealing with static networks where proactive protocols work at their best because of getting the picture of whole topology and independent of the data generation. Among the widely used proactive protocols; DSDV [23], FSR [24], and OLSR [26], we prefer DSDV because of the following reasons:

(A) ETX and ELP have been implemented in DSDV.

(B) FSR and OLSR only use periodic updates that consume more bandwidth due to large size. On the other hand, along with periodic updates, DSDV also uses trigger updates. The former ones carry all the available routing information or complete routing table, called 'full-dump'. While the later ones merely carry 'incremental'. An incremental is an information changed since last 'full-dump'and is well fitted in a Network Protocol Data Unit (NPDU). The trigger updates help DSDV to reduce routing overhead that raises throughput.

(C) Like [42] and [84], our implementation further enhances DSDV to never send 'full-dump' with trigger updates, called 'no-dumps', rather the full-dumps are sent merely at the 'full-dump' periods.

(D) FSR's 'graded frequency' mechanism along with the 'fish-eye' technique works better in dense networks and OLSR's Multi-point Relay (MPR) is suitable in static and dense networks. But for our simulations the case is contrary, as our network consists of 50 nodes.

(E) FSR and OLSR when receive any data packet, they immediately send it at the already calculated route. But on receiving a data packet, DSDV waits for a duration of WST (Weighted Settling Time) during which, if it finds some better route (provided by trigger or periodic update), it sends data on that route. This mechanism works well for quality link metrics.

In the absence of any mobility, IBETX achieves higher throughputs than the other three metrics chosen for the comparison; ETX, ETP and ELP, as shown in Fig.3. This performance is achieved due to implementing the multiple performance criteria in IBETX. In static wireless multi-hop networks, all nodes prefer the shorter paths and as a result the underlying network experiences congestion. Thus, to accurately measure the link quality is more important in static networks. Measuring the probability of success for data packet delivery using probes is more useful strategy when compared only with shortest path. So, the part ELD nicely performs the job and achieves higher throughput than ETX by avoiding the computational overhead of taking inverse of the probability of success $(d_f \times d_r)$

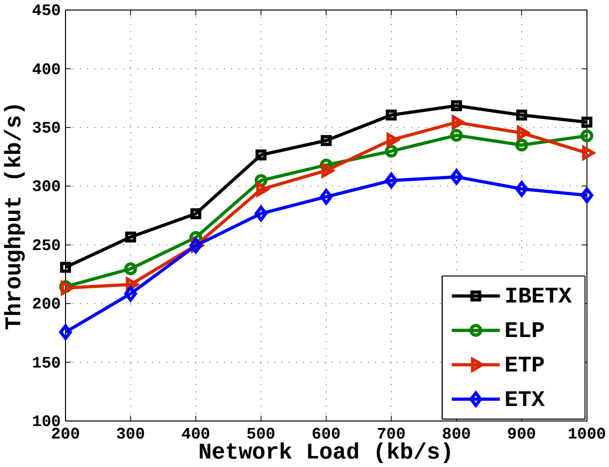


Figure 6.3: Comparison of avg throughput achieved by DSDV with four metrics

for all links, as shown in Fig.6.2.

But as the network under analysis is wireless by nature, the links with lower bit rate degrade the performance of the faster links. Therefore, taking the bandwidth of all links in the same contention domain into account gives more accurate information about the link status as compared to simple considering probability of success. In eq.(3), b_{exp} tackles this issue by implementing the bandwidth sharing mechanism of 802.11 DCF and considers the throughput reduction of faster links due to contention of slower links. Consequently, b_{exp} predicts quality links and helps IBETX to achieve increased throughput as compared to ELP, ETP and ETX, as obvious from Fig.6.3. The metric achieves 19% more throughput than ETX, 10% more than both ETP and ELP. As, probes are smaller in size as compared to data packets, so, the idea of measuring the link quality by calculating probability of success along with nominal bit rate information does not suffice. Therefore, our metric incorporates the interference part that rightly predicts the medium congestion and collisions due to hidden nodes that increase the end-to-end delay. To accurately estimate the medium occupation, using cross-layered approach, *ELI* periodically probes the MAC-layer 100 times per second. In MAC broadcast probes, all nodes in the network piggyback their interferences for the last $\tau_t seconds$ $(10s)$, hence, *ELI* avoids extra routing overhead. Moreover, since, IBETX can consider longer paths due to *ELB* and *ELI*, it increases throughput and reduces E2ED. So, our metric reduces end-to-end delay up to 15% lower than ETP, 16% lower than ELP and 24% lower than ETX; because only ELP directly implements interference but has some performance leaks, as not measuring bandwidth of the contending nodes and varying probe size, etc. Comparison of end-to-end delay produced by all of the competing four metrics is depicted in Fig.6.4.

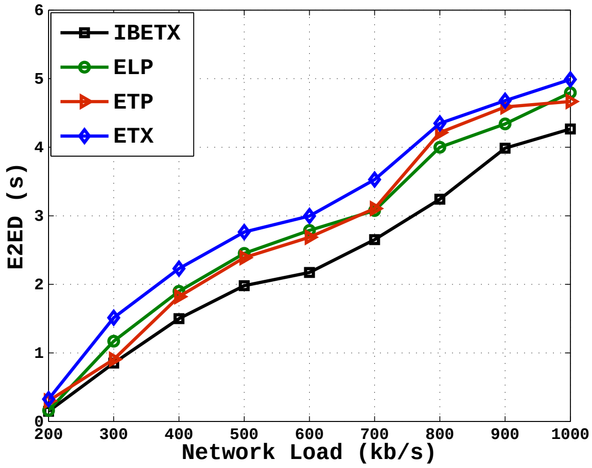


Figure 6.4: Average End-to-end delay produced by DSDV with four metrics

6.6 Conclusion

In this chapter, we proposed a new quality link metric for wireless multi-hop networks. We have overcome the performance leaks in ETX due to its unawareness from the MAC layer. Using cross-layer approach, we provided our metric with the MAC layer information. *ELD* found the high throughput paths more efficiently than ETX and ELP by avoiding the overhead due to computational complexities in both. *ELB* found the quality links from all active links in the same contention domain. *ELI* part along with *ELB* removed the deficiency in ETX and ETX based metrics to ignore the longer paths while selecting quality links, though the longer paths usually give higher throughputs.

Chapter 7

Conclusion

Contents

The work presented in this thesis has the objective to improve the performance of IEEE 802.11-based Wireless Multi-hop Networks (WMhNs). Particularly, selecting the quality routes and guaranteeing Quality of Service (QoS) in each flow. This chapter concludes the thesis with a general conclusion and the perspective of research for the future.

7.1 Conclusion

The massive simulations of the chosen protocols have demonstrated that reactive protocols are superior to the proactive ones, provided that mobility is taken into account as a constraint in the Wireless Multi-hop Networks (WMhNs). This can be concluded that AODV and DSR show the best performance during all mobilities and at all speeds. This study recommends that DSR can be selected for networks which are conscious about the number of hops, where the traffic is overhead-sensitive and nodes favor packet overhead on the low byte overhead. AODV should be chosen where the number of hops is not a problem and the nodes prefer low byte overhead on the packets. OLSR in proactive protocols is the plausible choice. Regarding the scalabilities and traffic flow analyses, detailed flow charts are generalized for both reactive and proactive protocols that provide clear ideas about the algorithms working behind and their flows. Among proactive protocols, DSDV best works for both scalabilities and traffic scenarios being simulated in this work. DSR, on average, over performs both AODV and DYMO for increasing number of nodes and number of packets, when throughput, end-to-end delay and routing overhead are considered.

After Minimum Hop-count link metric which usually selects lossy links, Expected Transmission Count (ETX) is the most widely used metric (in the presence of least mobility of nodes and availability of links). We therefore, analyzed and compared the performance of ETX-based routing metrics. Overheads occurred and throughputs achieved due to the factors added to ETX have been deeply analyzed. Along with the importance of a routing protocol, an issue of an efficiently designed routing link metric runs parallel. We therefore, have presented a comprehensive study on the design requirements for routing link metrics in a broader view. We discussed several possible issues regarding WMhNs that can better help in designing a link metric. The ambition of a high throughput network can only be achieved by targeting a concrete compatibility of the underlying wireless network, the routing protocol operating it, and routing metric; heart of a routing protocol. Depending upon the most demanding features of networks, different routing protocols impose different costs of 'message overhead' and 'management complexity'. These costs help to understand that which type of routing protocol is well suitable for which kind of underlying wireless network and then which routing link metric is appropriate for which routing protocol.

Based upon the above analyses, we proposed and implemented a new quality link metric for WMhNs. We have overcome the performance leaks in ETX due to its unawareness from the MAC layer. Using cross-layer approach, we provided our metric with the MAC layer information. With t he help of MAC layer, our proposed metric, IBETX selects the end-to-end paths with appreciable goodputs. *ELD* part of the metric found the high throughput paths more efficiently than ETX and ELP by avoiding the overhead due to computational complexities in both. The component *ELB* has chosen the quality links from all active links in the same contention domain. *ELI* part along with *ELB* removed the deficiency in ETX and ETX-based metrics to ignore the longer paths while selecting quality links, though the longer paths usually give higher throughputs.

7.2 Future Work

We proposed Interference and Bandwidth Adjusted (IBETX) routing metric, and implemented it in DSDV protocol. Comparative analysis of IBETX with different present metrics is made through a number of simulations in NS-2 that resulted in better performance of IBETX. However, we take small scale network for checking IBETX. Moreover, our scalability analysis over six protocols (AODV, DSDV, DSR, DYMO, FSR and OLSR) shows that DSDV outperforms other proactive protocols in small and medium node densities, and OLSR is one of the scalable protocols. Although, correct information of nominal bit rate from MAC layer is useful in single-channel multi-hop networks but nominal bit rate information becomes more useful for multi-channel networks. Therefore, in future, we are interested to enhance the functionality of IBETX to work in multi-channel environment. Moreover, because of the computational overhead reduction by *ELD*, IBETX can achieve even higher throughput values, if it is implemented with OLSR in network with more population of nodes.

Publications

[1] Nadeem Javaid, Ayesha Bibi, and Karim Djouani, ”Interference and Bandwidth Adjusted (ETX) in Wireless Multi-hop Networks”, IEEE Globecom Workshop SaCoNAS 2010.

[2] Dridi, K., Javaid, N., Daachi, B. and Djouani, K., ”IEEE 802.11 e-EDCF evaluation through MAC-layer metrics over QoS-aware mobility constraints”, 7th ACM MoMM, p-211-217, 2009.

[3] Khaled Dridi, Nadeem Javaid, Karim Djouani, and Boubaker Daachi, ”Performance Study of IEEE802.11e QoS in EDCF-Contention-Based Static and Dynamic Scenarios”, 2nd IEEE ICECS 2009.

[4] Javaid, N., Javaid, A., Khan, I.A., and Djouani, K., ”Performance study of ETX based wireless routing metrics”, 2nd IEEE IC4 2009.

[5] Nadeem Javaid, Khaled Dridi, Muti Ullah, and Karim Djouani, ”Identifying design requirements for wireless routing link metrics”, Submitted in ICTET 2011.

Bibliography

[1] Hamma, S. and Cizeron, E. and Issaka, H. and Guédon, J.P., "Performance evaluation of reactive and proactive routing protocol in IEEE 802.11 ad hoc network", hamma2006performance, 2006.

[2] Mbarushimana, C. and Shahrabi, A .,"Comparative study of reactive and proactive routing protocols performance in mobile ad hoc networks", mbarushimana2007comparative, 679-684, 2007.

[3] Clausen, T. and Jacquet, P. and Viennot, L., "Comparative study of routing protocols for mobile ad-hoc networks", clausen2002comparative, 2002.

[4] Das, S.R. and Castaneda, R. and Yan, J. and Sengupta, R ., "Comparative performance evaluation of routing protocols for mobile, ad hoc networks", das2002comparative, 153-161, 2002.

[5] Dyer, T.D. and Boppana, R.V., "A comparison of TCP performance over three routing protocols for mobile ad hoc networks", dyer2001comparison, 56-66, 2001.

[6] Cano, J.C. and Manzoni, P., "A performance comparison of energy consumption for mobile ad hoc network routing protocols", cano2002performance, 57-64, 2002.

[7] Broch, J. and Maltz, D.A. and Johnson, D.B. and Hu, Y.C. and Jetcheva, J., "A performance comparison of multi-hop wireless ad hoc network routing protocols", broch1998performance, 85-97, 1998.

[8] Das, S.R. and Perkins, C.E. and Royer, E.M., "Performance comparison of two on-demand routing protocols for ad hoc networks" das2002performance, 3-12, 2002.

[9] Chatzistavros, E. and Stamatelos, G. "Comparative performance evaluation of routing algorithms in IEEE 802.11 Ad Hoc networks", chatzistavros2009comparative, 19-24, 2009.

[10] Nitnaware, D. and Verma, A., "Performance Evaluation of Energy Consumption of Reactive Protocols under Self-Similar Traffic", nitnaware-performance

[11] U. Ashraf, S. Abdellatif and G. Juanole ìEfficient Route Maintenance in Wireless Mesh Networksî, Accepted in 3rd IEEE International Symposium on Wireless Pervasive Computing (IEEE ISWPC 2008) Santorini, Greece 7-9 May 2008.

[12] U. Ashraf, G. Juanole and S. Abdellatif ìEvaluating Routing Protocols for the Wireless Mesh Backboneî, In 3rd IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (IEEE WiMob2007) pp. 40-46, White Plains, New York, USA, October 2007.

[13] Tyagi, SS and Chauhan, RK, "Performance Analysis of Proactive and Reactive Routing Protocols for Ad hoc Networks", International Journal of Computer Applications IJCA, 31-34, 2010.

[14] Hu, Y.C. and Perrig, A. and Johnson, D.B., "Ariadne: A secure on-demand routing protocol for ad hoc networks", Wireless Networks, 21-38, 2005.

[15] Sham-ul-Arfeen et al, "Performance Evaluation of MANET Routing Protocols Using Scenario Based Mobility Models", Innovative Algorithms and Techniques in Automation, Industrial Electronics and Telecommunications, Springer, 2007. pp. 419-424.

[16] Wikipedia. "Orders of magnitude (speed)". Available May 27, 2010. http://en.wikipedia.org/wiki/Orders-of-magnitude-(speed).

[17] S. Das, et al., "Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks," in INFOCOM (1), 2000, pp. 3-12.

[18] C. Perkins, et al. IETF RFC3561, "Ad hoc On-Demand Distance Vector (AODV) Routing". Available: http://www.ietf.org/rfc/rfc3561.txt

[19] D. Johnson, et al., "DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks," in In Ad Hoc Networking, edited by Charles E. Perkins, Chapter 5, 2001, pp. 139-172.

[20] D. Johnson, et al., IETF RFC, "The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4"," 2007.

[21] I. Chakeres and C. E. Perkins, IETF Draft dymo-05, "Dynamic MANET On-demand (DYMO) Routing"," 2006.

[22] R. E. Thorup, " Implementing and Evaluating the DYMO Routing Protocol",Master Thesis, 2007.

[23] C. E. Perkins and P. Bhagwat, "Highly dynamic Destination-Sequenced Distance-Vector routing (DSDV) for mobile computers," SIGCOMM Comput. Commun. Rev., vol. 24, pp. 234-244, 1994.

[24] G. Pei, et al., "Fisheye State Routing in Mobile Ad Hoc Networks," in ICDCS Workshop on Wireless Networks and Mobile Computing, 2000, pp. D71-D78.

[25] M. Gerla, et al., IETF Draft-01, "Fisheye State Routing Protocol (FSR) for Ad Hoc Networks"," 2000.

[26] T. Clausen and P. Jacquet, "Optimized Link State Routing Protocol (OLSR)," ed, 2003.

[27] T. Clausen and P. Jacquet, IETF RFC-3626, "Optimized Link State Routing Protocol OLSR," The Internet Society http://www. ietf. org/rfc/rfc3626. txt, 2003.

[28] C. M. University, "AODV Copyright (c) 1997, 1998 Carnegie Mellon University, http://www.cmu.edu/," 1998.

[29] CMU/Monarch, "DSDV Copyright (c) 1997 Regents of the University of California, http://www.universityofcalifornia.edu/regents/ " 1997.

[30] CMU/Monarch, "DSR Copyright (C) 2000 by the University of Southern California. http://www.usc.edu/," 2000.

[31] University of Murcia,"MASIMUM (MANET Simulation and Implementation at the University of Murcia)" 2009.

[32] The Institute of Operating Systems and Computer Networks,"FSR". http://www.ibr.cs.tu-bs.de/ 2009.

[33] Q. Chen, et al., "Overhaul of ieee 802.11 modeling and simulation in ns-2," 2007, p. 168.

[34] Javaid, N., Javaid, A., Khan, I. A. and Djouani, K., "Performance Study of ETX Based Wireless Routing Metrics," 2nd International Conference on Computer, Control and Communication, pp. 1-7 (2009).

[35] Santivanez, C.A. and McDonald, B. and Stavrakakis, I. and Ramanathan, R.," On the scalability of ad hoc routing protocols ", p-1688-1697, 2002.

[36] Broch, J. and Maltz, D.A. and Johnson, D.B. and Hu, Y.C. and Jetcheva, J., "A performance comparison of multi-hop wireless ad hoc network routing protocols", p-85-97, 1998.

[37] Saleem, M. and Khayam, S.A. and Farooq, M.,"On performance modeling of ad hoc routing protocols", EURASIP Journal on Wireless Communications and Networking, 2010.

[38] Saleem, M. and Khayam, S.A. and Farooq, M.," A formal performance modeling framework for bio-inspired ad hoc routing protocols ",saleem2008formal, 103-110, 2008.

[39] Castaneda, R. and Das, S.R. and Marina, M.K., "Query localization techniques for on-demand routing protocols in ad hoc networks" 137-151, Wireless Networks ,2002.

[40] Park, I. and Pu, I., "Energy Efficient Expanding Ring Search", park2007energy, 198-199, 2007.

[41] Aguayo, D.S.J.D.C.D. and Morris, B.A.C.R., "Performance of Multihop Wireless Networks: Shortest Path is Not Enough", aguayo2002performance", 2002.

[42] D. S. J. de Couto, ”High-throughput routing for multi-hop wireless networks”, Ph.D. dissertation, MIT, 2004.

[43] Richard Draves, Jitendra Padhye, Brian Zill, ”Comparison of routing metrics for static multi-hop wireless networks”, Vol. 34, No. 4. (October 2004), pp. 133-144.

[44] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, ”Link-level measurements from an 802.11b mesh network,” in Proc. ACM SIGCOMM, Sep. 2004.

[45] C. E. Koksal and H. Balakrishnan, ”Quality-aware routing metrics for time-varying wireless mesh networks”, IEEE Journal on Selected Areas in Communications, vol. 24, no. 11, pp. 1984-1994, Nov. 2006.

[46] Miguel Elias M. Campista, Pedro Miguel Esposito, Igor M. Moraes, LuÌs Henrique M. K. Costa, and Otto Carlos M. B. Duarte, ”Routing Metrics and Protocols for Wireless Mesh Networks”.

[47] R. Draves, J. Padhye, and B. Zill, ”Routing in multi-radio, multi-hop wireless mesh networks”, in ACM International Conference on Mobile Computing and Networking (MobiCom), Sept. 2004, pp. 114-128.

[48] Yaling Yang, Jun Wang, and Robin Kravets, ”Interference-aware Load Balancing for Multihop Wireless Networks”, Tech. Rep. UIUCDCS-R-2005-2526, Department of Computer Science, University of Illinois at Urbana-Champaign, 2005.

[49] Y. Yang, J. Wang, and R. Kravets, ”Designing Routing Metrics for Mesh Networks”, WiMesh, 2005.

[50] Anand Prabhu Subramanian, Milind M. Buddhikot, and Scott Miller, ”Interference Aware Routing in Multi-Radio Wireless Mesh Networks”, Technical Report, Computer Science Department, Stony Brook University, 2007.

[51] Cunha, D.O. and Duarte, O. and Pujolle, G., ”An enhanced routing metric for fading wireless channels”, IEEE WCNC 2008.

[52] Weirong Jiang, Shuping Liu, Yun Zhu and Zhiming Zhang, ”Optimizing Routing Metrics for Large-Scale Multi-Radio Mesh Networks”, p-1550-1553, WiCom 2007.

[53] J. C. Park and S. Kasera,"Expected Data Rate: An Accurate High-Throughput Path Metric For Multi-Hop Wireless Routing", in proc. of IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, SECON 2005, Santa Clara, California, USA, Sep 2005.

[54] Vivek P. Mhatre, Henrik Lundgren, Christophe Diot, "MAC-Aware Routing in Wireless Mesh Networks". p-46-49, WONS 2007.

[55] P. Kyasanur and N. H. Vaidya, "Routing and Link-layer Protocols for Multi-Channel Multi-Interface Ad hoc Wireless Networks", Mobile Computing and Communications Review, 10(1): pp. 3 1-43, Jan. 2006.

[56] B. Awerbuch, D. Holmer, and R. Rubens. The Medium Time Metric, "High throughput route selection in multi-rate ad hoc wireless networks", Springer Mobile Networks and Applications, vol. 11, no. 2, pp. 253-266, April 2006.

[57] Daniel Aguayo, John Bicket, and Robert Morris. Srcrr: "A high throughput routing protocol for 802.11 mesh networks (draft)", Internet Article: http://pdos.csail.mit.edu/ rtm/srcrr-draft.pdf, 2006.

[58] ChenWang, Guokai Zeng, and Li Xiao, "Optimizing End to End Routing Performance in Wireless Sensor Networks", Springer-Verlag DCOSS 2007, LNCS 4549, pp. 36-49, 2007.

[59] Q. Dong, S. Banerjee, M. Adler, and A. Misra. "Minimum energy reliable paths using unreliable wireless links", In Proc. of ACM MobiHoc, 2005.

[60] Sabyasachi Roy, Dimitrios Koutsonikolas, Saumitra Das, and Y. Charlie Hu, "High-Throughput Multicast Routing Metrics in Wireless Mesh Networks".

[61] Das, S.M. and Pucha, H. and Papagiannaki, K. and Hu, Y.C., "Studying wireless routing link metric dynamics", 7th ACM SIGCOMM conference on Internet measurement, p332, 2007.

[62] Yang, Y. and Wang, J., "Design guidelines for routing metrics in multihop wireless networks", IEEE INFOCOM, 2008.

[63] Hassanein, H. and A. Zhou, ”Routing with Load Balancing in Wireless Ad Hoc Networks”. Proc. ACM MSWiM, Rome, Italy, pp: 89-96. 2003.

[64] Wu, K. and J. Harms, ”Load Sensitive Routing for Mobile Ad Hoc Networks”. Proc. IEEE ICCCN, Phoenix, AZ, 2001.

[65] Lee, S.J. and M. Gerla, ”Dynamic Load Aware Routing in Ad Hoc Networks”. Proc. ICC, Helinski, Finland, pp: 3206-3210. 2001.

[66] Ahn, S., Y. Lim and J. Choe, ”A Load-Balancing Approach in Ad-Hoc Networks”. ICOIN, LNCS 2662, pp: 672-681. 2003.

[67] J.H. Song, V.W.S. Wong and V.C.M. Leung, ”Efficient On-Demand Routing for Mobile Ad-Hoc Wireless Access Networks”. IEEE GLOBECOM, pp: 558-563. 2003.

[68] Saigal, V., A.K. Nayak, S.K. Pradhan and R. Mall, ”Load Balanced Routing in Mobile Ad hoc Networks”. Elsevier Computer Communications, 2004.

[69] Yoo, Y. and S. Ahn, 2004. ”A Simple Load- Balancing Approach in Secure Ad Hoc Networks”. ICOIN 2004.

[70] Song, J.H., V. Wong and V.C.M. Leung, ”Load-Aware On-Demand Routing (LAOR) Protocol for Mobile Ad hoc Networks”. The 57th IEEE Semiannual Vehicular Technology Conference, p3: 1753-1757.

[71] Hongqiang Zhai and Yuguang Fang, ”Impact of Routing Metrics on Path Capacity in Multirate and Multihop Wireless Ad Hoc Networks”, Proceedings of the IEEE International Conference on Network Protocols, 2006.

[72] Boughanmi, N. and Song, Y.Q., ”A New routing metric for satisfying both energy and delay constraints in wireless sensor networks”. Journal of Signal Processing Systems for Signal, Image, and Video Technology, p137–143, 2008.

[73] O. Chipara, Z. He, G. Xing, Q. Chen, X. Wang, C. Lu, J. Stankovic, and T. Abdelzaher, ”Real-time Power-Aware Routing in Sensor Networks”, Forteenth IEEE International Workshop on Quality of Service, 2006.

[74] J. Tsai and T. Moors, "A Review of Multipath Routing Protocols: From Wireless Ad Hoc to Mesh Networks", ACoRN Early Career Researcher Workshop on Wireless Multihop Networking, 2006.

[75] Faheem, Y. and Rougier, J.L., "Loop avoidance for fish-eye OLSR in sparse wireless mesh networks", 6th international conference on Wireless On-Demand Network Systems and Services, p215-218, 2009.

[76] W. Cordeiro et al.," Providing quality of service for mesh networks using link delay measurements", 16th International Conference on Computer Communications and Networks, 2007.

[77] D. Passos, D. V. Teixeira, D. C. Muchaluat-Saade, L. C. S. Magalhaes, and C. V. N. de Albuquerque, "Mesh network performance measurements", I2TS, 2006.

[78] Moreira, W., Aguiar, E., AbelÈm, A., Stanton, M., "Using multiple metrics with the optimized link state routing protocol for wireless mesh networks", SimpÛsio Brasileiro de Redes de Computadores e Sistemas DistribuÌdos, Maio (2008).

[79] Hongqiang Zhai and Yuguang Fang, "Impact of Routing Metrics on Path Capacity in Multirate and Multihop Wireless Ad Hoc Networks", Proceedings of the IEEE International Conference on Network Protocols pp. 86-95, 2006.

[80] M. Kodialam and T. Nandagopal, "Characterizing the Capacity Region in Multi-Radio Multi-Channel Wireless Mesh Networks", Proc. ACM Mobi-Com'05, Sept. 2005.

[81] K. Paul, S. Bandyipadhyay, A. Mukherjee, and D. Saha, "Communication-aware mobile hosts in ad-hoc wireless network", IEEE ICPWC,1999.

[82] K. Scott and N. Bambos, "Routing and channel assignment for low power transmission in PCS", In ICUPC, vol. 2, pp. 498-502, 1996.

[83] J-P. Sheu, C-T. Hu, and C-M. Chao, "The Handbook of Ad Hoc Wireless Networks, Chapter Energy-Conserving Grid Routing Protocol in Mobile Ad Hoc Networks", RCR Press LLC, 2003.

[84] Usman A. *et al.*, "An Interference and Link-Quality Aware Routing Metric for Wireless Mesh Networks,". IEEE 68th Vehicular Technology Conference, 2008.