L1 ① Large Scale Path Loss (Signal Attenuation 衰减)

dB (decibel) 1W = 30dBm = 0dBW 3dB = 2 times

Average received signal power S: $S = kr^{-a}$

$S_{dB} = 10\log_{10} kr^{-a} = 10[\log k - a\log r]$

k: constant
r: transmitter–receiver distance
a: signal attenuation / path loss exponent
a = 2 in free space

Multipath Propagation: reflection, Diffraction (衍射), scattering (散射)

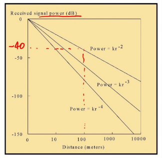
Maximum doppler shift: $f_d = v/\lambda = \frac{vf}{c}$

Received envelope $p(r) = \frac{r}{\sigma^2}\exp\left[-\frac{r^2}{2\sigma^2}\right]$

mean: $\bar{r} = E(r) = 1.2533\sigma$

mean square: $E(r^2) = 2\sigma^2$

variance: $\sigma_r^2 = 0.4292\sigma^2$



Level Cross Rate

- The ***level crossing rate*** is the no. of times per second the signal envelope fades below the ***fade margin***, *A*. The level crossing rate is a 2nd-order statistic given by:

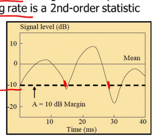
$$N_A = \int_0^\infty \dot{r}p(r = A, \dot{r})d\dot{r} = \sqrt{2\pi}f_d\frac{A}{\sqrt{2}\sigma}\exp\left[-\frac{A^2}{2\sigma^2}\right]$$

where $\dot{r} = dr/dt$, and $p(r = A, \dot{r})$ is the joint distribution

$$p(r,\dot{r}) = \frac{r}{\sqrt{2\pi\nu^2}\sigma^2}\exp\left[-\frac{1}{2}\left(\frac{r^2}{\sigma^2} + \frac{\dot{r}^2}{\nu^2}\right)\right]$$

and ν^2 is the variance of the derivative of the quadrature components, i.e.

$\nu^2 = E\{\dot{A}_I^2(t)\} = E\{\dot{A}_Q^2(t)\} = 2\pi^2 f_d^2\sigma^2$



- The ***fade margin*** is the minimum received signal level for successful recovery of the information/message
- Suppose f_d = 10 and $A^2/(2\sigma^2) = 0.1$, then $N_A = 7.17$
- Suppose f_d = 10 and $A^2/(2\sigma^2) = 0.01$, then $N_A = 2.48$

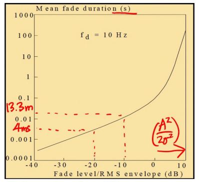
normalized fade margin

CK Tham, ECE NUS 20

- Average fade duration in seconds (time that envelop is < *A*) is:

$$\bar{t}_F = \frac{P(r \le A)}{N_A} = \frac{1}{\sqrt{2\pi}f_d}\frac{\sqrt{2}\sigma}{A}\left[\exp\left(\frac{A^2}{2\sigma^2}\right) - 1\right]$$

- Suppose f_d = 10 and $A^2/(2\sigma^2) = 0.1$, then $\bar{t}_F$ = 13.3 ms
- Suppose f_d = 10 and $A^2/(2\sigma^2) = 0.01$, then $\bar{t}_F$ = 4.0 ms

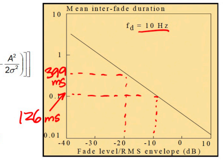


- Ave. inter-fade duration in seconds (time that envelope is > *A*) is:

$$\bar{t}_I = \frac{1}{N_A} - \bar{t}_F = \frac{1}{N_A} - \frac{P(r \le A)}{N_A} = \frac{1}{N_A}[1 - P(r \le A)]$$
$$= \frac{1}{\sqrt{2\pi}f_d\frac{A}{\sqrt{2}\sigma}\exp\left(-\frac{A^2}{2\sigma^2}\right)}\left[1 - \left[1 - \exp\left(-\frac{A^2}{2\sigma^2}\right)\right]\right] = \frac{1}{\sqrt{2\pi}f_d}\frac{\sqrt{2}\sigma}{A}$$

where $P(r \le A)$ is the prob. that the envelope $r \le A$, is:

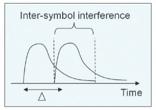
$$P(r \le A) = \int_0^A p(r)dr = \int_0^A \frac{r}{\sigma^2}\exp\left(-\frac{r^2}{2\sigma^2}\right)dr = 1 - \exp\left(-\frac{A^2}{2\sigma^2}\right)$$



- Suppose f_d = 10 and $A^2/(2\sigma^2) = 0.1$, then = 126 ms
- Suppose f_d = 10 and $A^2/(2\sigma^2) = 0.01$, then = 399 ms

As no. of scatterers increases, discrete multipath merge into a continuous pulse of length Δ, commonly known as the delay spread.

Delay spread limits data rate to below $1/\Delta$ to avoid ***inter-symbol interference***, beyond which special measures are required to overcome data error.



maximum symbol rate $= \frac{1}{\Delta}$

Ratio of signal energy per bit (E_b) to noise power density per Hertz (N_0):

$$\frac{E_b}{N_0} = \frac{S/R}{N_0}$$

Bit Error Rate (BER) for digital data transmission is a function of E_b/N_0

- as bit rate *R* increases, transmitted signal power *S* must increase to maintain required E_b/N_0

DSDV: Destination Sequenced Distance - Vector

- Based on the Distance Vector (Distributed Bellman-Ford) algorithm
  - modified to work better in MANET
  - produces loop-free fewest hops path
- Each node maintains a routing table containing:
  - next hop towards each destination
  - a cost metric (distance in no. of hops) for the path to each destination
  - a destination sequence number that is created by the destination (sequence numbers help to distinguish new routes from stale ones, thereby avoiding formation of loops)
- Each node periodically forwards its routing table to its neighbors
  - each node increments and appends its own sequence number when sending its local routing table
  - this sequence number will be attached to route entries created for this node

L2: ALOHA

$T_{send} = T_{tom} + 2T_{prop}$ (T_{tom} = time to clock out data packet / T_{prop} = propagation delay from base to satellite)

Total delay from sending packet to receiving acknowledgement: $T_{send+ack} = 2[T_{tom} + 2T_{prop}] + T_{response}$ | contention window: $2T_{tom}$

probability of successful transmission $P(0) = e^{-2T_{tom}\lambda}$ λ = arrival rate $\lambda_{max} = \frac{1}{2T_{tom}}$

maximum throughput: $S_{max} = \frac{1}{2T_{tom}e}$ packets/s ($S = \lambda P(0) = \lambda e^{-2T_{tom}\lambda}$) $P(0)_{min} = \frac{1}{e}$

traffic carried per terminal: $A = \lambda \cdot T_{tom}$ max: $A_{max} = 0.5$ erlang

traffic carried max $= S_{max} \cdot T_{tom} = \frac{1}{2e} = 0.184$ erlang

retransmission traffic $= A_{max} - \frac{1}{2e} = 0.316$

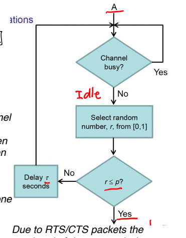
Slotted ALOHA: $S_{max} = \frac{1}{e}$ $P(0) = e^{-T_{tom}\lambda}$ $P(0)_{min} = \frac{1}{e}$ Average attempts $= \frac{1}{P(0)}$

Carrier Sense Multiple Access (CSMA)

p-Persistent CSMA

idle 空闲

τ = max propagation delay between 2 stations
p = specified probability



Advantages of persistent CSMA:
The terminal will keep listening to the medium when it senses the medium busy. Then it will capture the channel immediately after the medium becomes idle. Thus, 1-persistent CSMA will have good channel utilization when traffic is light. The p-persistent CSMA works better when traffic is heavy.
Disadvantages of persistent CSMA:
When traffic is heavy, 1-persistent will bring more collisions because it is highly possible that more than one terminal sense the medium and immediately transmits after the medium becomes idle.
Advantages of non-persistent CSMA:
It can reduce the collisions by waiting for a random time to sense the medium again after it senses the medium is busy. It is especially useful when traffic is heavy.
Disadvantages of non-persistent CSMA:
It may waste the channel by waiting for an unnecessary random time to sense the medium again due to light traffic.
Therefore, choose 1-persistent CSMA when traffic is light, and non-persistent CSMA or p-persistent CSMA when traffic is heavy.

Due to RTS/CTS packets the overhead of data transmission increases. If we have small DATA packets then it is not worth using the RTS/CTS packets when increase the overhead. However, if the DATA packets are bigger in size and the channel is heavily loaded then using RTS/CTS can prove effective.

DIFS: Distributed coordination function Inter-Frame Space

SIFS: Short Inter-Frame Space

Hidden node:

Exposed node:

CSMA-CA

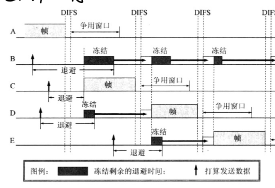
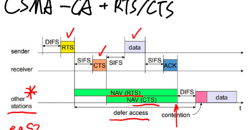


图 9-7 802.11 的退避机制的概念

CSMA-CA + RTS/CTS

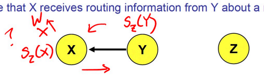


e.g. S2

why SIFS is kept smaller than the DIFS period?
This is to give priority to some packets. For example after transmitting the DATA packets the receiver should be allowed to send ACK packets without any collisions. To accomplish this, we transmit the ACK packets immediately after SIFS time interval before any other node can transmit and interrupt the ACK packet.
SIST can't be 0. SIFS is the turnaround time required to change from transmit (receive) mode to receive (transmit) mode. Thus, it is compulsory to have finite SIFS time.

What is the earliest time for the receiver to send the CTS message?
DIFS + (RTS + Prop.) + SIFS
what is the shortest time for the receiver to send the ACK signal?
DIFS + (RTS + Prop.) + SIFS + (CTS + Prop.) + SIFS + (Data + Prop.) + SIFS

Assume that X receives routing information from Y about a route to Z



- S(X): destination sequence # for node Z as stored at node X
- S(Y): destination sequence # for node Z sent by Y with its routing table to node X

Node X takes the following steps:

- If S(X) > S(Y), then X ignores the routing information received from Y
- If S(X) = S(Y), and cost of going through Y is smaller than the route known to X, then X sets Y as the next hop to Z
- If S(X) < S(Y), then X sets Y as the next hop to Z, and S(X) is updated to equal S(Y)

DSR: Dynamic Source Routing
① routing discovery
② routing maintenance

①Source broadcasts a packet containing address of source and destination

If the packet has been received before by a mobile node, discard it 丢弃

Each node checks whether it knows of a route to the destination

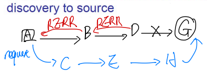
If it does not, it adds its own address to the packet and then forwards the packet

If the destination has a route to the source in its route cache, use it

Otherwise, if symmetric links are supported, reverse the route in the ***route record*** RERR

If symmetric links are not supported, the destination initiates its own route discovery to source

Route cache (A): A B D G X / A C E H G



L3 Routing

Dijkstra's Algorithm Method

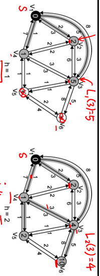
- Step 1 [Initialization]
  - *T* = {*s*} Set of nodes so far incorporated consists of only source node
  - *L*(*n*) = *w*(*s*, *n*) for *n* ≠ *s*
  - Initial path costs to neighboring nodes are simply link costs
- Step 2 [Get Next Node]
  - Find neighboring node *x* not in *T* with least-cost path from *s*
  - Incorporate node *x* into *T*
  - Also incorporate the edge that is incident on node *x* and a node in *T* that contributes to the path
- Step 3 [Update Least-Cost Paths] n is dest under consideration
  - *L*(*n*) = min[*L*(*n*), *L*(*x*) + *w*(*x*, *n*)] for all *n* ∉ *T*
  - If latter term is minimum, path from *s* to *n* is path from *s* to *x* concatenated with edge from *x* to *n*
- Algorithm terminates when all nodes have been added to *T*

Iteration	T	L(2)	Path	L(3)	Path	L(4)	Path	L(5)	Path	L(6)	Path
1	{1}	2	1–2	5	1-3	1	1–4	∞	-	∞	-
2	x=4 {1,4}	2	1–2	4	1-4-3	1	1–4	2	1-4–5	∞	-
3	x=2 {1, 2, 4}	2	1–2	4	1-4-3	1	1–4	2	1-4–5	∞	-
4	x=5 {1, 2, 4, 5}	2	1–2	3	1-4-5–3	1	1–4	2	1-4–5	4	1-4-5–6
5	x=3 {1, 2, 3, 4, 5}	2	1–2	3	1-4-5–3	1	1–4	2	1-4–5	4	1-4-5–6
6	x=6 {1, 2, 3, 4, 5, 6}	2	1-2	3	1-4-5-3	1	1-4	2	1–4–5	4	1-4-5-6

Bellman-Ford Algorithm

- Step 1 [Initialization]
  - $L_0(n) = \infty$, for all $n \ne s$
  - $L_h(s) = 0$, for all *h*
- Step 2 [Update]
- For each successive *h* > 0 h=1,2, ..
  - For each *n* ≠ *s*, compute
  - $L_h(n) = \min_j[L_{h-1}(j) + w(j,n)]$
- Connect *n* with predecessor node *j* that achieves minimum path cost
- Eliminate any connection of *n* with different predecessor node formed during an earlier iteration
- Path from *s* to *n* terminates with link from *j* to *n*

cost $L_h(j)$



h	$L_h(2)$	Path	$L_h(3)$	Path	$L_h(4)$	Path	$L_h(5)$	Path	$L_h(6)$	Path
0	∞	-	∞	-	∞	-	∞	-	∞	-
1	2	1-2	5	1-3	1	1-4	∞	-	∞	-
2	2	1-2	4	1-4-3	1	1-4	2	1-4-5	10	1-3-6
3	2	1-2	3	1-4-5-3	1	1-4	2	1-4-5	4	1-4-5-6
4	2	1-2	3	1-4-5-3	1	1-4	2	1-4-5	4	1-4-5-6

Bellman-Ford algorithm: [distance-vector protocol]

- Each node bases its path calculation on knowledge of the cost to all directly connected neighbors plus the advertised costs for routes heard from these neighbors.
- Easily susceptible to loops because the routers depend on information from neighbors, which might no longer be useful after a failure
- Has a distributed version that is fairly effective.

Dijkstra's algorithm: [link-state protocol]

- Requires each node to have complete information about the entire topology.
- Each router requires more memory to hold the link-state database and more processing capacity to run the algorithm.
- Used in routing algorithms such as Open Shortest Path First (OSPF) (for wired networks) and Optimized Link State Routing (OLSR) (for mobile ad hoc networks).

Latency of route discovery

- Proactive protocols may have lower latency since routes are maintained at all times
- Reactive protocols may have higher latency because a route from X to Y will be found only when X attempts to send to Y

VS

Overhead of route discovery/maintenance

- Reactive protocols may have lower overhead since routes are determined only if needed
- Proactive protocols can result in higher overhead due to continuous route updating

AODV: Ad Hoc On-demand Distance Vector Routing (for dynamic networks)

- AODV includes route discovery and route maintenance
- AODV minimizes the number of broadcasts by creating routes on-demand
- AODV only supports the use of symmetric links because a RREP (Route Reply) is forwarded along the path established by a RREQ (Route Request)
- AODV uses Hello messages to maintain the local connectivity of a node

maintenance:

- If the source node moves, it re-initiates route discovery to establish a new route to the destination.
- If intermediate node moves, its upstream node sends a Route Error (RERR) message to the source.
- When a source receives the RERR, it will reinitiate the route discovery if the route is still needed.

Comparison of AODV & DSR

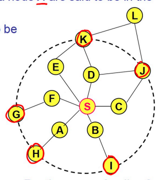
- DSR has less routing overhead than AODV.
- DSR is based on a source routing mechanism, whereas AODV uses a combination of DSR and DSDV mechanisms.
- AODV has better performance than DSR in higher-mobility scenarios.
- DSR has less frequent route discovery processes than AODV.
- after route expires, AODV needs to discover route again

AODV vs DSR when request

By using DSR algorithm to create a route from nodes 6 to 23, node 6 at first checks its route cache to determine whether it already has a route to the destination 23, if it has, it will use this route. If it does not have such a route, it initiates route discovery by broadcasting a route request packet. This route request contains the address of node 23. A reply is generated when the route request reaches either node 23, or an intermediate node whose route cache contains an unexpired route to the destination.
For AODV, node 6 broadcasts a route request packet (RREQ) to its neighbours, which then forwards the request to their neighbours, and so on, until either node 23 or a node with a fresh enough route to node 23 is located. The main difference between the DSR and AODV is that DSR uses source routing and AODV uses forwarding tables at each node. During the process of forwarding the RREQ, nodes record in their route tables the address of the neighbour from which the first copy of the broadcast packet is received, thereby establishing a reverse path.

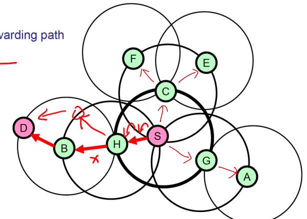
ZRP: Zone Routing Protocol

- ZRP combines
  - Proactive protocol: which pro-actively updates network state and maintains route regardless of whether any data traffic exists or not
  - Reactive protocol: which only determines route to a destination if there is some data to be sent to the destination
- All nodes within hop distance at most *d* from a node *X* are said to be in the routing zone of node *X*
- All nodes at hop distance exactly *d* are said to be peripheral nodes of node *X*'s routing zone 边缘
- Intra-zone routing: pro-actively maintain state information for links within a short distance from any given node
  - Routes to nodes within short distance are thus maintained proactively.
- Inter-zone routing: use a route discovery protocol for determining routes to far away nodes. Route discovery is similar to DSR with the exception that route requests are propagated via peripheral nodes.



Routing zone of radius 2

Indicating the forwarding path
S → H → B → D



ZRP combines intra-zone and inter-zone routing. It defines a zone of distance d such that nodes within the zone use pro-active routing by maintaining state information for links between them, and inter-zone routing uses a route discovery protocol for determining routes beyond the existing zone. Route discovery is similar to DSR with the exception that route requests are propagated via peripheral nodes. Peripheral nodes lie at the boundary between two or more zones and they act as intermediate nodes between zones and relay packets across zones.

TCP/IP

Describe how indirect TCP deals with error-prone wireless channels.
Indirect-TCP partitions the TCP connection to 2 parts, one for the wired network and another to the wireless network. This separates the ability of traditional TCP to deal with packet loss/congestion in wired network from the adversities of wireless channel that traditional TCP is inadequate. The separation to 2 connections allows the basestation (BS) and the MH to deploy a link-aware lightweight transmission protocol to efficiently and quickly carry out packet transmissions between the BS and MH. This is a workaround to achieve the need for end-to-end reliable transmission.
The original paper on Indirect-TCP by A Burke and BR Badrinath focused on enhancing handoff performance. It deployed one of the standard mobile-IP solutions, namely the Columbia mobile-IP protocol, over the wireless link to deal with a mobile device that is changing association with different base stations (or gateways) that have different IP addresses. If a host moves from one network to another - migrates - either the host should change its address to fit the new network or the routing mechanisms should be able to forward the messages into a network where the recipients address is an alien. The internet routing scheme was originally created at a time when the mobility of hosts was not considered to be an issue, thus all routing mechanisms are more or less static and cannot handle misrouting of individual addresses to be made in the existing TCP in order for it to be used in a wireless medium.

What are the disadvantages of using wireline TCP over wireless networks?
The main disadvantage of using wireline TCP over wireless networks is that wireline TCP attributes loss of packets during packet transmission to congestion in the network. However, this may not be the case in wireless networks, where packet losses occur mainly due to the physical nature of the medium such as attenuation, thermal effects and interference in the air medium. The wireline TCP thus goes into congestion control mechanism in these cases when there is no need to do this. This further reduces the throughput.

What are the inherent characteristics of wireless networks that require changes in existing TCP?
High packet error rate, high latency, frequent link breakages due to mobility, and device handoffs are some of the inherent characteristics of wireless networks which require changes to be made in the existing TCP in order for it to be used in a wireless medium.

Indirect TCP

- Indirect-TCP is a split-connection solution.
- It suggests that any TCP connection from a mobile host (MH) to a fixed host (FH) in the wireline network should be split into two separate connections.
- One of the connections is between the MH and the base station (BS) over the wireless medium.
- The other connection is between the BS and the FH over the wireline network.
- A packet which is sent to the MH is first received by the BS.
- BS then sends an acknowledgement to the FH and forwards the packet to the MH.
- If the MH moves to a different cell while communicating with a FH, the whole connection information maintained at the current BS is transferred to the new BS. handoff
- The FH is unaware of this redirection and is not affected when this switch occurs.
- Since the end-to-end connection is split, the TCP connection over the wireless link can handle wireless related errors and handoff disruptions.
- Indirect-TCP separates the congestion control functionality on the wireless link from that on the wireline network, which enables the two kinds of links to identify different reasons for packet losses and then take corresponding actions.
- Since the TCP connection is broken into two, it is possible for a mobile host to use some lightweight transport protocol instead of a full TCP/IP suite to communicate with the base station and access the wireline network through the base station.
- This feature is desirable since a mobile host has limited battery and processing power.
- However, Indirect-TCP violates the end-to-end semantics of TCP acknowledgements, as both the wireline and wireless part of a connection have their own acknowledgements.
- The control overhead is also considerable as the base station needs to maintain a significant amount of state information for each TCP connection and all the state information needs to be transferred to the new base station in the event of a handoff, which could lead to long delay.

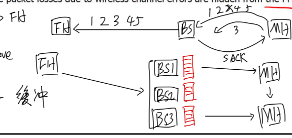
SNOOP

- SNOOP sought to improve TCP performance by modifying the network-layer software at a BS while preserving end-to-end TCP semantics.
- It adds a Snooping module to the network layer, which monitors every packet that passes a BS in either direction.
- If TCP packets are sent from a FH to a MH, the SNOOP module caches each packet that has not yet been acknowledged by the MH.
- Meanwhile, the SNOOP module also keeps track of all acknowledgements sent from the mobile host.
- The SNOOP module determines that a packet loss occurs by detecting if it receives a duplicate acknowledgement or its local timer times out.
- In this case, the lost packet is retransmitted if it has been cached.
- The duplicate acknowledgements, if any, are suppressed.
- In this way, unnecessary congestion control mechanism invocations are avoided since packet losses due to wireless channel errors are hidden from the FH.

MH → FH

MH move

buffer 缓冲



Wireless Mesh network

Architecture
Nodes are comprised of mesh routers and mesh clients

Mesh routers:

- Conventional wireless AP (Access Point) functions
- Additional mesh routing functions to support multihop communications
- Usually multiple wireless interfaces built on either the same or different radio technologies

Mesh clients:

- Can also work as a router for client WMN
- Usually only one wireless interface

Classification of WMN architecture:

- 1. Infrastructure/Backbone WMNs
- 2. Client WMNs
- 3. Hybrid WMNs

A $\xrightarrow{d_f = 0.2}$ B, $d_r = 0.2$

WMN metrics:

① Expected Transmission Count (ETX) $ETX = \frac{1}{d_f \cdot d_r}$

② Expected Transmitting Time (ETT) $ETT_i = ETX_i \frac{S}{B_i}$ S: packet size, B_i: the data rate in i-th link

③ Weighted cumulative expected transmission time (WCETT)

$$WCETT_p = \underbrace{(1-\beta)\sum ETT_i}_{\text{end to end delay}} + \underbrace{\beta \max_{1\le j\le k} X_j}_{\text{channel diversity along path}}$$

$0 \le \beta \le 1$ tunable parameter
X_j: represent the sum of ETT for the hops that are in the channel j

④ Airtime Ca $C_a = \left[O + \frac{B_t}{r}\right]\left[\frac{1}{1-e_f}\right]$

O: channel access overhead
B_t: test frame size by bytes
r: data rate in Mb/s
e_f: measured test frame error rate

Advantages:

Very low installation and maintenance cost

- No wiring! Wiring is always expensive/labour intensive, time consuming, inflexible

Easy to provide coverage in outdoors and hard-to-wire areas:

- Ubiquitous access

Rapid deployment

Self-healing, resilient, extensible, scalable

Problems/Difficulties
Latency is a big issue in WMN!
- multi-hop
- collisions
- interference

How CSMA-CA works

The CSMA-CA MAC works as follows:

Part 1 – Before Transmitting

- If channel is idle for more than a Distributed coordination function Inter-Frame Space (DIFS) time and the backoff counter is zero, a station can transmit immediately
- If channel is busy, the station generates a random backoff period that is uniformly distributed from [0 to current Contention Window (CW) size.]
- At the initial backoff stage, the current CW is set at the minimum contention window (CW_{min}) size.
- Backoff counter decrements by 1 if channel is idle for each time slot and freezes if channel is sensed busy.
- Backoff counter is re-activated to count down when channel is sensed idle for more than DIFS time.
- If backoff counter reaches 0, station will attempt to transmit.

Part 2 – After Transmitting

- If successful, the destination station sends an acknowledgement (ACK) after a Short Inter-Frame Space (SIFS) time and the current CW size is reset to CW_{min}.
- If a station does not receive an acknowledgement within an acknowledgement timeout period after a frame is transmitted, i.e. it is not successful, it will increase the current contention window CW size by doubling it plus one (until a maximum contention window (CW_{max}) size is reached) in the next backoff stage and a new random backoff period is selected as before.
- This process repeats itself until the packet is successfully transmitted or until the maximum retry limit is reached. If the packet is still not successfully transmitted, then it is dropped.

RM-AODV

1. On-demand Routing
- Based on Radio Metric AODV (RM-AODV): use the features of AODV (RFC 3561) but uses Radio-aware path selection with arbitrary path metrics; Destinations may be discovered in the mesh on-demand.

2. Pro-active Routing
- Based on tree-based routing: If a Root portal (=MPP) is present, a distance vector routing tree is built and maintained
- Tree based routing is efficient for hierarchical networks
- Tree based routing avoids unnecessary discovery flooding during discovery and recovery

1. On-demand Routing in HWMP – Key Features

- Allows nodes to quickly obtain routes for new destinations (does not require nodes to maintain routes to destinations that are not in active communication)
- Route Discovery
  - Uses expanding ring search to limit the flood of routing packets
  - Reverse Paths are setup by *Route Request* packets broadcast (or unicast) from Originator
  - Forward Paths are setup by *Route Reply* packet sent from destination node or any intermediate node with a valid route to the destination

Route Maintenance

- Nodes monitor the link status of next hops in active routes. When a link break in an active route is detected, a *Route Error* message is used to notify other nodes that the loss of that link has occurred.
- Route Error message is a unicast message, resulting in quick notification of route failure

Loop Freedom

- DSDV and AODV principle
- All nodes in the network own and maintain a destination sequence number which guarantees the loop-freedom of all routes towards that node

Pro-active

2. Tree-based routing in HWMP – Key Features

•Topology Creation

downstream
- Root MP may issue a "broadcast" RREQ; MPs may respond with RREP

upstream
- The Root MP may issue "Root Announcements" which contains information on path metrics to Root MP
- MPs may respond by a unicast RREQ to the Root (answered by RREP) (to establish bi-directional path)
- MPs select next hop to Root based on best path metric
- "Registration" of subtrees by MPs facilitates outward message routing
- *Drawback*: Packets between MPs are routed through Root MP eventhough a shorter path may exist

•Topology Maintenance

- MPs monitor their upstream links and may switch to back-up links using RREP; This avoids "re-building" the tree. Loss of upstream link X causes RERR to be sent down.
- This allows nodes to decide/select own back-up paths. It signals route holders that some route is broken.