Unit-5

Wireless ATM and Ad Hoc Routing

**ATM**

In this section, a brief introduction to ATM is made in order prior to discussing Wireless ATM. ATM, also known as cell-relay for reasons that will be described later, is a technology capable of carrying any kind of traffic, ranging from circuit-switched voice to bursty data, at very high speeds. ATM possesses the ability to offer negotiable QoS. Thus, ATM is the technology of choice for multimedia networking applications that demand both large band-widths and QoS guarantees since these properties cannot typically be offered by conventional networks such as Ethernet LANs.

ATM is a packet-switching technology that somewhat resembles frame relay. However, the main difference is the fact that ATM has minimal error and flow control capabilities in order to reduce control overhead and also that ATM utilizes fixed-size (53 bytes) packets known as cells instead of variable-sized packets as in frame relay. Fixed size packets enable fast speeds for ATM switches and together with the reduced overhead give rise to the very high data rates offered by ATM.

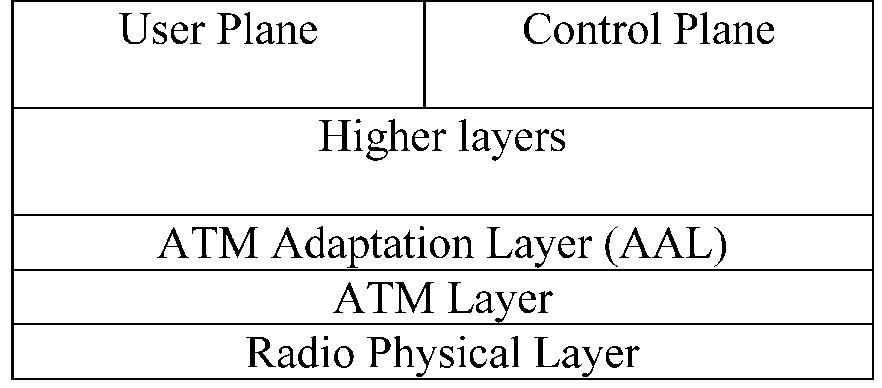


Figure: ATM protocol architecture

The ATM protocol architecture is shown in Figure 10.1. Its main parts are:

* Physical layer. It involves the specification of the transmission medium and the signal encoding to be used. The two alternative speeds offered by the physical layer are 155 and 622 Mbps.
* ATM layer. This defines the transmission of ATM cells and the use of connections either between users, users and network entities or between network entities. These connections are referred to as Virtual Channel Connections (VCCs) and are analogous to the data link connections in frame relay. VCCs can carry both user traffic and signaling information. A collection of VCCs that share the same endpoints is known as a Virtual Path Connection (VPP).
* The ATM Adaptation Layer (AAL). This layer maps the cell format used by the ATM layer to the data format used by higher layers. Thus, at the transmitting side, AAL maps frames coming from higher layers to ATM cells and hands them over to the ATM layer for transmission. On the receiving side, ATM reassembles cells into the respective frames and passes frames to upper layers. A number of AALs exist, each of which corresponds to a specific traffic category. AAL0 is virtually empty and just provides direct access to the cell relay service. AAL1 supports services that demand a constant bit rate, which is agreed during connection establishment and must remain the same for the duration of the connec-tion. This category of service is known as Constant Bit Rate (CBR) service with typical examples being voice and video traffic. AAL2 supports services that can tolerate a variable bit rate but pose limitations regarding cell delay. This category of service is known as Variable Bit Rate (VBR) service with typical examples being transmission of compressed (e.g. MPEG) video where bit rate can vary, however, delay guarantees are needed to avoid jerky motion. AAL3/4 and AAL5 support variable-rate traffic with no delay requirements. Such categories are VBR traffic with no delay bounds, Available Bit Rate (ABR), which is a best effort service that guarantees neither rate nor delay but only minimum and maxi-mum rate and Unspecified Bit Rate (UBR) which is essentially ABR without a minimum rate guarantee.

The protocol architecture shown in Figure 10.1 also defines three separate planes. These are:

1. the user plane, which provides for transfer of user information and associated control information (e.g. FEC, ARQ); (b) the control plane, which performs call control and connec-tion control; and (c) the management plane, which includes plane management for manage-ment of the whole system and coordination of the planes and layer management for management of functions relating to the operation of the various protocol entities.

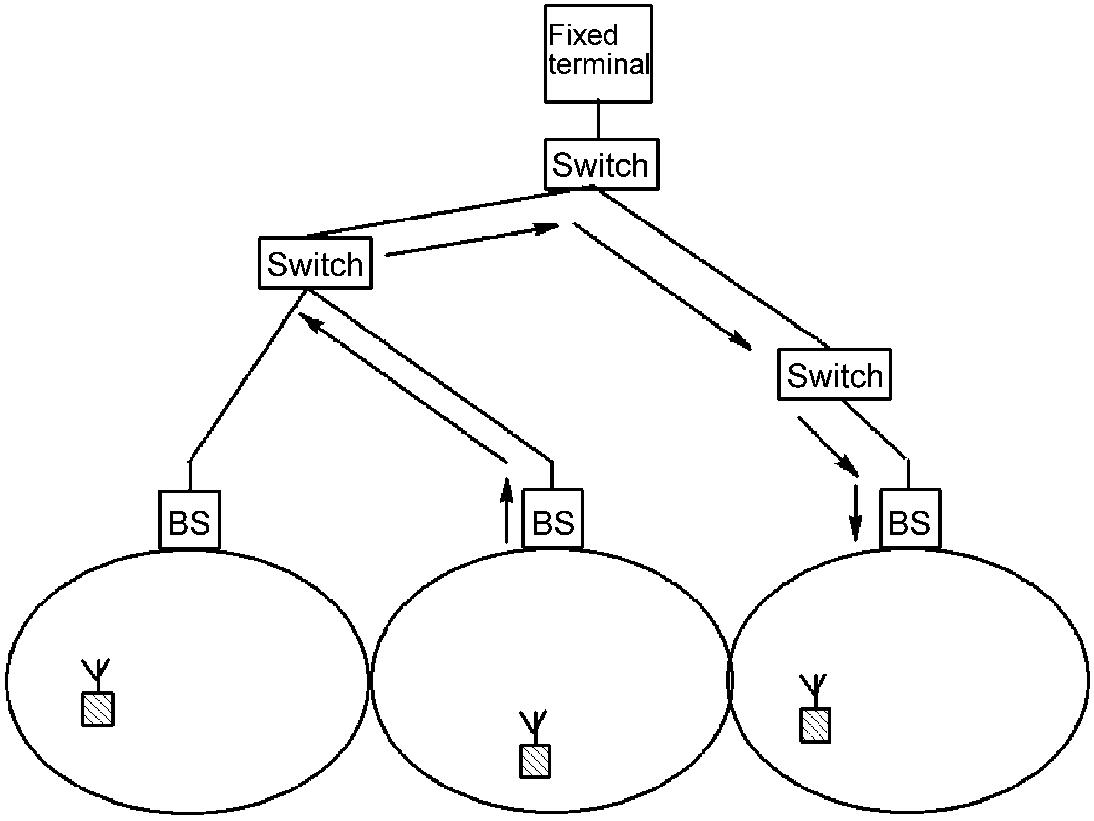


Figure:WATM network architecture

**Wireless ATM**

A simple network architecture for WATM is shown in Figure 10.2. It consists of a number of small cells, each of which contains a BS. The basic role of the base station is interconnection of the wireless and wired segment of the network. Each cell contains a number of mobile ATM-enhanced terminals. All terminals inside a cell communicate only with the cell’s BS and not between each other. To support mobile terminals, BSs are connected to mobility-enhanced ATM switches. These in turn are interconnected by regular switches in the ATM backbone network. ATM switching is used for intercell traffic. Terminals are capable of roaming between cells and this gives rise to the need for techniques for efficient location management and efficient handoff.

There are proposals for two different scenarios [5] regarding the functionality of the BS in the above architecture. The first scenario calls for termination of the ATM Adaptation Layer (AAL) at the BS. In this case, the traffic transmitted over the wireless medium is not in the format of ATM cells. Rather a custom wireless MAC is used that encapsulates one or more ATM cell into a single packet.

ATM implementation over the wireless medium poses several design and implementation challenges that are summarized below:

* ATM was originally designed for a transmission medium whose BERs are very low (about 10210). However, wireless channels are characterized of low bandwidth and high BER values. It is questioned whether ATM will function properly over such noisy transmission channels.
* ATM calls for a high resource environment, in terms of transmission bandwidth. However,

as we have seen, the wireless medium is a scarce resource that calls for efficient use of medium. However, an ATM cell carries a header, which alone poses an overhead of about 10%.

**Wireless ATM Architecture**

The protocol architecture currently proposed by the ATM Forum is shown in Figure 10.3. The WATM items are divided into two parts: mobile ATM, which consists of a subpart of

the control plane, and radio access layer (shaded items in the figure). Mobile ATM deals with the higher-layer control/signaling functions that support mobility. The radio access layer is responsible for the radio link protocols for wireless ATM access. Radio access layers consists of the physical layer, the media access layer, the data link layer, and the radio resource control. Up to now, only PHY and MAC are under consideration. The protocols and approaches for DLC and RRC have not been proposed yet. The physical, MAC and DLC layers for the radio access layer are briefly discussed below, while mobile ATM issues are discussed in later sections.

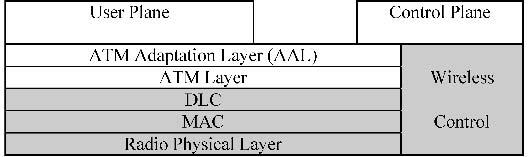


Figure : WATM protocol architecture

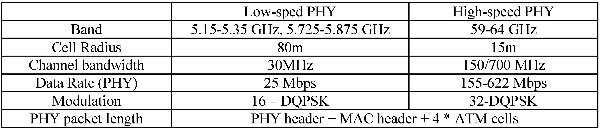


Figure: Physical layer requirements for WATM

**The Radio Access Layer**

**Physical Layer (PHY)**

Fixed ATM stations can typically achieve rates ranging from 25 to 155 Mbps at the PHY layer. However, due to the use of the wireless medium, such speeds are difficult to achieve in WATM. Thus, typical bit rates for WATM PHY are in the region of 25 Mbps, corresponding to the 25 Mbps UTP PHY option for wired ATM. Note that 25 Mbps is the speed at the physical layer. WATM VCs will typically enjoy bit rates ranging from 2 to 5 Mbps sustained and from 5 to 10 Mbps peak. Nevertheless, higher PHY speeds are possible and WATM projects under development such as the MEDIAN project succeeded in achieving data rates of 155 Mbps by employing OFDM transmission at 60 GHz.

**MAC Layer**

A number of MAC protocols have been proposed for WATM [5,7]. Most of the proposals describe a form of a centralized TDMA system in which the frames are divided into two parts: one contention part, which is used by the mobiles to reserve bandwidth for transmission and one part in which information is transmitted.

Some general requirements for an efficient WATM MAC protocol are the following [5]:

* Allow for decreased complexity and energy consumption at the mobile nodes.
* Provide a means of supporting negotiated QoS under any load condition.
* Support the standard ATM services, such as UBR, ABR, VBR and CBR traffic classes.
* Provide adequate support for QoS-demanding traffic classes.
* Provide a low delay mechanism of channel assignment to connections.

**DLC Layer**

The DLC layer interfaces the ATM layer to lower layers. Thus, in order to hide the deficien-cies of the wireless medium from the ATM layer, DLC should implement error detection, retransmission and FEC. Different levels of coding redundancy might be used in order to support each ATM service class.

The DLC layer exchanges 53-byte ATM cells with the ATM layer above it. A DLC PDU is a packet that may consist of one or more cells. This packet is handed down to the physical layer for transmission as a single unit. The use of a multicell DLC packet reduces overhead but requires functionality to convert between the ATM cell format and the DLC packet format.

**Mobile ATM**

**Location Management/Connection Establishment**

Existing protocols for connection setup in ATM assume that the location of a terminal is fixed. Thus, the terminal’s address can be used to identify its location, which is needed in processes such as call establishment. However, when terminals become mobile, this is no longer true and additional addressing schemes and protocols are needed to track the mobile ATM terminal.

Location management in a wireless ATM network can be either external to the connection procedure or integrated[3,8]. Here we describe the latter option. Each mobile terminal served by the network is associated with a ‘home’ BS or switch which provides it with a home ATM address. When the terminal moves to another cell, it is assigned a foreign address via this cell’s BS. The home switch maintains a pointer from the permanent home address to the current foreign address of the mobile.

**Handover in Wireless ATM**

The mobility nature of terminals in WATM networks means that the network must be able to dynamically switch ongoing connections of users that roam between cells. Handovers take place when mobiles move out of the coverage of a BS towards the coverage of a new one. In such a case the signal measurement at the mobile of the new BS gets stronger while that of the previous one weakens. Handoff can be network-controlled, mobile-assisted or mobile-controlled. In the first case, the mobile terminal is completely passive and all signal measure-ments and handoff initiations are a responsibility of the BS. In the second case, both the BS and the mobile terminal perform signal measurements, however, the handoff initiation is a responsibility of the BS. Finally, in the third case both the BS and the mobile terminal perform signal measurements and the handoff initiation is a responsibility of the mobile terminal.

A handover should be done in an efficient way such that the user does not notice perfor-mance degradation. Of course, there is a chance of the handoff being blocked. This means that the new BS is not able to serve the connections of the roaming user, either for reasons of bandwidth availability or due to the fact that it cannot guarantee the QoS of the user’s connection. In the latter case, however, a renegotiation towards a lower level of QoS might be carried out in order for the connection to be kept alive.

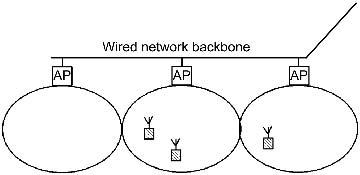
* The terminal initiates the handoff. This is done by sending a message to its current BS in order to initiate the procedure of moving the connection from the current BS to the new one.
* The network switches and BSs collectively determine the switch from which to reroute each VC. This switch is known as a ‘crossover switch’ (COS). When the handover occurs, the current QoS may not be supported by the new data path. In this case, a negotiation is required to set up new QoS. Handover algorithms should take those facts into considera-tion. Related to this fact is the identification of the optimal COS to be used in order to switch the connection. COs may be initiated either at the old or the new BS.
* Upon determination of the COS, the network routes a subpath from the COS to the new BS.
* Over the above path, the cell stream is switched to the new BS.
* The unused subpath from the COS to the old BS is released.
* Finally, the terminal drops its radio connection with the old BS, connects to the new one and confirms end-to-end handoff.

**HIPERLAN 2: An ATM Compatible WLAN**

HIPERLAN 2 aims to provide high speed access (up to 54 Mbps at the physical layer) to a variety of networks including 3G networks, ATM and IP based networks and for private use as a wireless LAN system. Supported applications include data, voice and video, with specific QoS parameters taken into account. In contrast to the WLAN systems described in Chapter 9, HIPERLAN 2 is a connection-oriented system which uses fixed size packets. HIPERLAN 2 is compatible with ATM. Its connection-oriented nature makes support for QoS applications easy to implement. In the following subsections, we describe the main aspects of HIPERLAN 2.

**Network Architecture**

The HIPERLAN 2 standard adopts an infrastructure topology. As shown in Figure 10.5, the network coverage area comprises a number of cells, with traffic in each cell being controlled by an Access Point (AP). Mobile terminals within a cell communicate with the cell’s AP through the HIPERLAN 2 air interface. Direct communication between two mobile terminals is also possible, however. this procedure is still in the development phase. Each mobile terminal can communicate only with one AP (that of the current cell). In order for such a communication to take place, an association procedure must first take place between the AP



**Figure: HIPERLAN 2 network architecture**

and the mobile terminal. After the association takes place, mobile terminals can freely move within the coverage area of the HIPERLAN 2 network while maintaining their logical connections. Moving to another cell is made possible through a handover procedure. The APs automatically configure the network by taking into account changes in topology due to mobility. Association and handover are revisited later in this section.

Being compatible with ATM, HIPERLAN 2 is a connection-oriented network using fixed size packets. Signaling functions are used to establish connections between the mobile nodes and the AP in a cell and data is transmitted over these connections as soon as they are established, using a time division multiplexing technique. The standard supports two types of connections: bi-directional point-to-point connections between a mobile node and an AP, and unidirectional point-to-multipoint connections carrying traffic to the mobile nodes. Finally, there is a dedicated broadcast channel used by the AP to transmit data to all mobiles within its coverage.

The connection-oriented nature of HIPERLAN 2 makes support for QoS applications easy to implement. Each connection can be created so as to be characterized by certain quality requirements, like bounded delay, jitter and error rate. This support enables the HIPERLAN 2 network to support multimedia applications in a way similar to the ATM network.

HIPERLAN 2 also provides support for issues like encryption and security, power saving, dynamic channel allocation, radio cell handover, power control, etc. However, most of these issues are either not standardized yet or left to the vendors to implement.

**The HIPERLAN 2 Protocol Stack**

The protocol stack for the HIPERLAN 2 standard is shown in Figure 10.6. It comprises a control plane part and a user plane part following the semantics of ISDN functional partition-ing. The user plane includes functionality for transmission of traffic over established connec-tions, and the control plane provides procedures to control established connections. The protocol has three basic layers: the Physical Layer (PHY), the Data Link Control (DLC) layer, and the Convergence Layer (CL). At the moment, only the DLC includes control plane functionality. The various layers are discussed below.

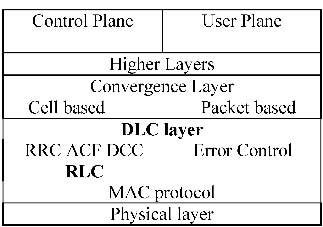


Figure: The HIPERLAN 2 protocol stack

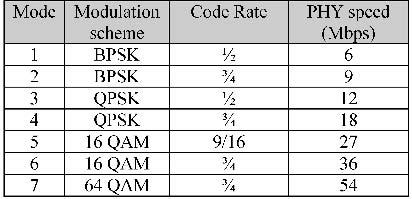


Figure:HIPERLAN 2 physical layer alternatives

**HIPERLAN 2 Physical Layer**

HIPERLAN 2 is characterized by high transmission rates at the physical layer, up to 54 Mbps. The use of OFDM in the physical layer effectively combats the increased fading occurrence experienced in indoor radio environments, such as offices, etc., where the transmitted radio signals are subject to reflection from a number of objects, thus leading to multipath propaga-tion and consequently ISI. The channel spacing is 20 MHz with 52 subcarriers used for each channel. Of these, 48 subcarriers carry actual data and the remaining four are used as pilots in order to perform coherent demodulation.

HIPERLAN 2 is able to adapt to changing radio link quality through a Link Adaptation (LA) mechanism. Based on received signal quality which depends both on the AP-mobile terminal relative position and interference from nearby cells, LA dynamically selects the method of modulation and the Forward Error Correction (FEC) code to use in an effort to provide a robust physical layer. The alternative modulation methods are BPSK, QPSK, 16 QAM and 64 QAM. FEC is performed by a convolutional code with rate 1/2 and constraint length 7. The physical layer alternatives offered by LA are shown in Figure 10.7.

**HIPERLAN 2 Data Link Control (DLC) Layer**

The DLC layer is used to establish the logical links between APs and the MTs. The DLC layer comprises a number of sublayers providing medium access and connection handling services to upper layers. The DLC layer consists of three sublayers: the Medium Access Control (MAC) sublayer, the Error Control (EC) sublayer and the Radio Link Control (RLC) sublayer.

MAC Protocol and Channel Types The MAC protocol used by HIPERLAN 2 is based on time-division duplex (TDD) and dynamic time-division multiple access (TDMA). MAC control is centralized and performed by each cell’s AP. The wireless medium is shared in the time domain through the use of a circulating MAC frame containing slots dedicated either to uplink or downlink traffic. The length of the MAC frame is fixed at 2 ms and comprises a number of parts which are not fixed. Rather, their lengths are variable in nature and are determined by the AP. Uplink and downlink slots within a frame are allocated dynamically depending on the need for transmission resources. All data from both mobile terminals and APs is transmitted in dedicated time slots. For mobile terminal

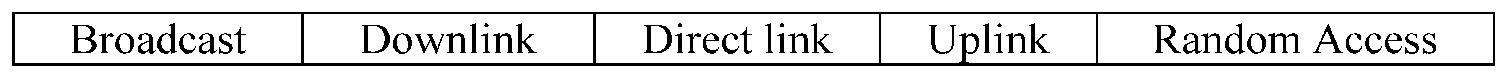


Figure: Structure of the 2 ms MAC frame

transmission, slots are allocated after bandwidth requests made to the AP. The exact form of the MAC frame is shown in Figure, where one can see that apart from the parts dedicated to uplink and downlink traffic there are also broadcast, direct link and random access phases. The broadcast frame carries the broadcast control channel and the frame control channel (both are described below). The direct link phase enables exchange of user traffic between mobile terminals without intervention of the AP. As mentioned above, this is optional. Finally, the random access phase carries the random access channel (described below). This phase is used by mobile terminals either for purposes of association with an AP, for control signaling when the terminal has not been allocated uplink slots within the MAC frame and during handover to a new AP for the purpose of switching ongoing connections to the new AP.

The MAC frame consists of several transport channels:

* The Broadcast Channel (BCH) is a downlink channel used to convey to the mobiles control information regarding transmission power levels, wake-up indicators for nodes in power save mode, length of the FCH and the RCH channels (described below) and the means to identify the HIPERLAN 2 network and the AP to which the mobile belongs.
* The Frame Control Channel (FCH) is a downlink channel used to notify the mobile nodes about resource allocation within the current MAC frame both for uplink and downlink traffic and for the RCH.
* The Random Access Channel (RCH) is used in the uplink both in order to request trans-mission in the downlink and uplink portions of future MAC frames and to transmit signaling messages. The RCH comprises contention slots which are used by the mobiles to compete for reservations. Collisions may occur and the results from RCH access are reported back to the mobiles in the Access Feedback Channel (ACH). When the request for transmission resources from the MTs arise, the AP can allocate more resources for the RCH in order to serve the increased demand.
* The Access Feedback Channel (ACH) is used on the downlink to notify about previous access attempts made in the RCH.

The above transport channels are used as a means to support a number of logical HIPERLAN 2 channels. The mapping is shown in Figures 10.9 and 10.10. The logical channels are as follows:

* The Slow Broadcast Channel (SBCH). All nodes within a cell can access the SBCH. It is a

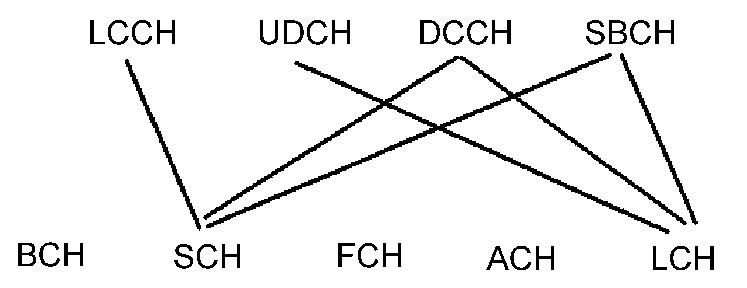


Figure: Mapping from logical to transport channels (downlink)

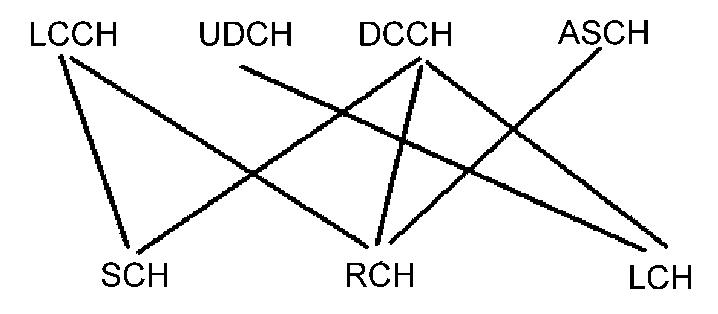


Figure: Mapping from logical to transport channels (uplink)

downlink channel that conveys broadcast control information concerning all the nodes within a cell. This transmission is initiated upon decision of the AP and may contain information regarding (a) the seed to be used for encryption, (b) handover acknowledg-ments, (c) MAC address assignments to non-associated mobile terminals, and (d) broad-cast of RLC and CL information.

* The Dedicated Control Channel (DCCH) is of bidirectional nature and is implicitly estab-lished when a terminal associates with the AP within a cell. After association with an AP has taken place, a terminal has its dedicated DCCH which is used to convey control signaling. The DCCH is realized as a DLC connection upon which RLC messages regard-ing association and control of DLC connections are exchanged.
* The User Data Channel (UDCH) transports user data cells between a mobile node and an AP and vice versa. A UDCH for a specific mobile node is established through signaling transmitted over the node’s DCCH. The UDCH establishment takes place after negotiation of certain quality parameters that characterize a connection. The DLC guarantees in-sequence delivery of the transmitted data cells to the convergence layer. The use of ARQ techniques is possible in UDCH operation, although there might be connections where ARQ is not used, such as multicasts and broadcasts. For uplink traffic, mobile requests for UDCH bandwidth are conveyed to the AP which then notifies the mobile whether or not it has been granted bandwidth through the FCH. For downlink traffic, the AP can reserve UDCH bandwidth without requests from mobiles.
* The Link Control Channel (LCCH) is a bidirectional channel used to exchange informa-tion regarding error control (EC) over a specific UDCH. The AP determines the necessary transmission slots for the LCCH in the uplink and the grant is announced in an upcoming FCH.
* The Association Control Channel (ASCH) is used by the mobile nodes either to request association or disassociation from a cell’s AP. Such messages are exchanged only (a) when a mobile terminal de-associates with an AP and (b) when a handover takes place.

Routing in Wireless Ad Hoc Networks:

A brief introduction to packet routing in wireless ad hoc networks was made in Chapter 2. There, it was highlighted that the performance of such protocols largely depends on the efficiency of the routing protocol used. In wireless ad hoc networks, stations are free to move around. This, together with the fact that the transmission range of mobile terminals is fixed, results in a dynamically changing network topology: As stations move around, some network links are destroyed while the possibility of new links being established arises. It is obvious that such an environment cannot be served efficiently by routing protocols developed for wired networks. This is due to the fact that in such networks, the assumption of a static topology is made. Thus, new routing protocols are needed for the dynamically changing ad hoc wireless environment.

This section describes some representative routing protocols for ad hoc wireless networks.

In these protocols, it is assumed that all stations of the network have identical capabilities and employ the capability to perform routing-related tasks, such as route discovery/establishment to other nodes in the network and route maintenance. The routing protocols presented fall into two families: table-driven and on-demand [12].

Table-driven routing protocols aim to maintain consistent, up-to-date routing information from each node to all other nodes of the network. Thus, each network node maintains one or more routing table which is used to store the routes from this node to all other network nodes. This knowledge regarding every possible route needs to be present in every node irrespective of the fact that some of these routes may not be used by network connections. When topo-logical changes occur, the relating information is relayed to all network nodes in an effort to provide the network nodes with up-to-date routing information.

On-demand routing protocols follow a different approach: a route is established only when required for a network connection. Thus, when a source node A needs to connect to a destination node B, then A invokes a routing discovery protocol to find a route connecting it to B. Upon route establishment, nodes A and B as well as intermediate nodes store the information regarding the route from A to B in their routing tables. The route is maintained until the destination is unreachable or the route is no longer needed.

Table-driven routing protocols obviously have the advantage of reduced end-to-end delay, since, upon generation of a network connection request, the route is already established.

**Table-driven Routing Protocols**

**Destination-Sequenced Distance-Vector (DSDV) Routing Protocol**

The DSDV routing protocol is an extension of the classical Bellman–Ford routing algorithm. The extensions incorporated in DSDV target freedom from loops in routing tables. In DSDV, each node maintains a routing table that contains information regarding all possible routes within the network, the number of hops of each route and the sequence number of each route. The latter is a number assigned by the destination of the route and shows how ‘old’ the route is. The lower the sequence number, the ‘older’ the route. When a node A needs to select a route to node B, it checks its routing table. If more than one such route is found, the newer one (the one with the largest sequence number) is used. If more than one route shares the same sequence number, then the shortest one (the one with the lower number of hops) is chosen.

Network nodes periodically broadcast their routing tables in order to propagate topology knowledge throughout the network. Apart from these periodic transmissions, a station can select to broadcast its routing table when significant topology changes have occurred. The propagation of routing tables obviously results in a large overhead. In an effort to alleviate this problem, two types of updates are defined: full-dump updates and incremental updates. In full-dump updates, stations transmit their entire routing table. Since routing tables are mostly quite large, a full-dump update typically involves more than one packet broadcast. This obviously consumes resources, so full dumps are transmitted infrequently. Incremental updates are transmitted between full dumps and convey only that information which has changed since the last update. Incremental updates thus consume less resources and are carried over a single packet. The relative frequency of full-dump and incremental updates depends on the nature of topological changes. In a network of a slowly changing topology, full dumps are rarely used since incremental dumps are able to convey the slow topological changes. On the other hand, in a network of fast changing topology, full dumps will be more frequent.

**Cluster head Gateway Switch Routing (CGSR) Protocol :**

CGSR is a modification of DSDV. It is different from DSDV in that, while DSDV assumes a ‘flat’ network (which means that all nodes have identical responsibilities), CGSR partitions the network into a number of ‘clusters’. Nodes inside a cluster are controlled by a node known as the clusterhead. Clusterheads are selected by the members of each cluster. It is obvious that as mobile nodes move, some clusters will disappear, new ones will be created and new nodes may be admitted into existing clusters. Thus, new clustherhead selections will appear from time to time. In an effort to reduce the overhead due to clusterhead selections, a Least Cluster Change (LCC) clusterhead selection algorithm is used. LCC states that clusterhead selections take place only when two clusterheads come into transmission range of one another or when a node moves out of the range of all the clusterheads.

CGSR uses a modification of DSDV as the routing scheme. Specifically, in CGSR all routes commencing from nodes inside a certain cluster pass through this cluster’s clusterhead. If a route serves a connection between two nodes inside the same cluster, then the clusterhead routes packets of this connection to their destination. If the route serves a connection between nodes in different clusters, then the clusterhead routes this packet to a gateway node. These are the nodes that are within range of more than one clusterhead. Upon receipt of the packet by the gateway, this is routed to the clusterhead of the adjacent cluster. The procedure continues until the packet reaches the clusterhead of its destination. Then, it is routed to the destination station. An example of CGSR routing is shown in Figure 10.12.

In GGSR, nodes maintain two tables: The routing table and the ‘cluster member table’. The ‘cluster member table’ contains the clusterhead of each node in the network. These tables are periodically transmitted by each node. Upon receipt of such a table from a neighbor, network nodes update their own ‘cluster member table’. ‘Cluster member tables’ are needed for packet routing. Upon reception of a packet, a node will check its cluster member table to find the clusterhead of the next cluster along the route to the destination station. Then, it checks its routing table to find the next hop that should be selected to reach the next clusterhead and forwards the packet over this hop.

**The Wireless Routing Protocol (WRP):**

In order for WRP to operate, each node must maintain four tables, a fact that can lead to substantial memory requirements, especially in the case of networks comprising many nodes. the four tables are the distance table, the routing table, the link-cost table and the Message Retransmission List (MRL) table. For a node A, the distance table of A contains the distance

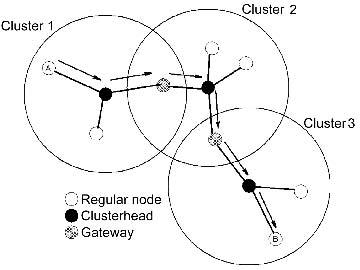


Figure: CGSR routing from node A to node B

to each destination node X via each neighbor Y of A. Moreover, each entry contains the downstream neighbor of Y through which the route from A to X traverses. The routing table of node A contains the distance to each destination node X, successor of A in this route and a flag that indicates whether this route is a simple one, or a loop. The link cost table of node A maintains the cost of the link from A to each neighbor Z and the number of timeouts since an error-free message was received from Z. Finally, the MRL contains entries regarding update messages sent from A. Such an entry comprises the sequence number of the update message, a retransmission counter, a flag indicating whether an acknowledgement is required from the neighbor for an update transmitted by A and a list of updates sent in the update message. Thus, the information in the MRL contains information regarding (a) neighboring nodes that have not acknowledged update messages from A, (b) when to retransmit update messages to these nodes.

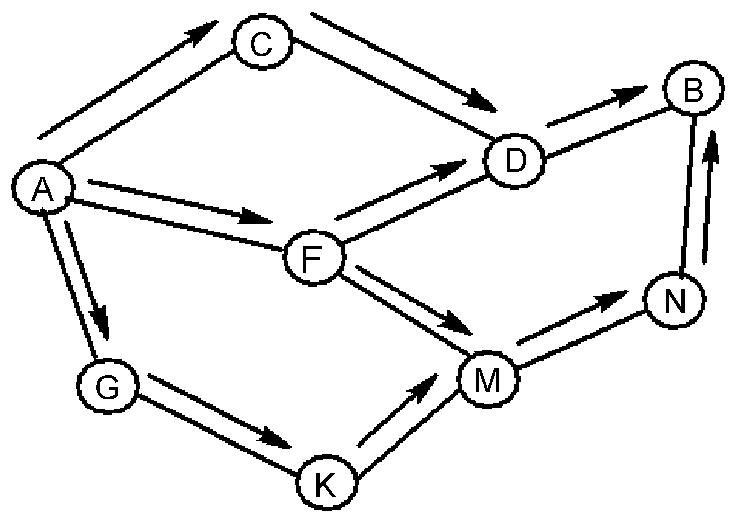
In WRP, nodes exchange update messages with their neighbors both periodically and as a result of link changes. Such is the case of a link loss between two nodes, e.g. A and B. In such cases, A and B send update information to their neighbors, which in turn modify their tables and search for alternative routes that do not contain the link between A and B. Updates contain information regarding new route destinations (that may have been established by neighboring nodes and other nodes in the network), new distances of routes, the predecessor of each route’s destination and a list of nodes that should acknowledge this update.

Figure: Propagation of the RREQ packet

**On-demand Routing Protocols**

Ad hoc On-demand Distance Vector (AODV) Routing

The AODV algorithm is the on-demand counterpart of the table-based DSDV algorithm. Their primary difference lies in the fact that AODV creates routes on-demand while DSDV maintains the list of all the routes. In AODV, a route is created only when requested by a network connection and information regarding this route is stored only in the routing tables of those nodes that are present in the path of the route.

The procedure of route establishment is shown in Figures 10.13 and 10.14. In this example, we assume that node A wants to set up a connection with node B. In Figure 10.13, node A initiates a path discovery process in an effort to establish a route to node B, by broadcasting a Route Request (RREQ) packet to its immediate neighbors. Each RREQ packet is identified through a combination of the transmitting node’s IP address and a broadcast ID. The latter is used to identify different RREQ broadcasts by the same node and is incremented for each RREQ broadcast. Furthermore, each RREQ packet carries a sequence number (similar to that of DSDV) which allows intermediate nodes to reply to route requests only with up-to-date route information. Upon reception of an RREQ packet by a node, this is forwarded to the immediate neighbors of the node and the procedure continues until the RREQ is received either by node B or by a node that has recently established a route to node B. If subsequent copies of the same RREQ are received by a node, these are discarded.

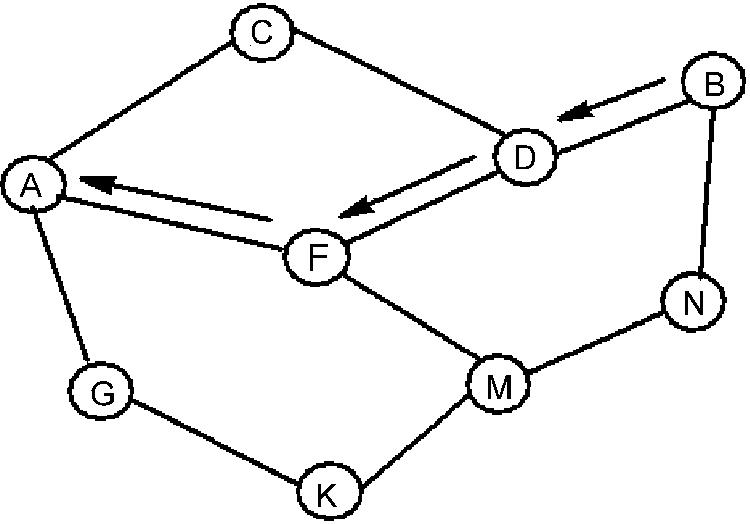


Figure: Propagation of the RREP packet

When a node forwards a RREQ packet to its neighbors, it records in its routing tables the address of the neighbor node where the first copy of the RREQ was received. This fact helps nodes to establish a reverse path, which will be used to carry the response to the RREQ. Returning to the previous example, we see in Figure 10.14 that when the RREQ has reached its destination, a route reply packet is sent back to A. Notice that the RREP follows the route B–D–F–A due to the fact that the first reception of the RREQ packet from B was due to node D and the first reception of the RREQ packet from D due to node F. As the RREP packet travels along the reverse path, the nodes that constitute the path (D, F, A) make appropriate changes in their routing tables (pointing to the next neighbor that is a part of this route) which identify the forward path from A to B. Due to the fact that the RREP packet travels along the reverse path traveled by the RREQ, AODV supports only the use of symmetric links. Support for asymmetric links is not provided. Upon establishment of a route, each route entry at each node is associated with a ‘lifetime’ value. A timer starts running when the route is not used. If the timer exceeds the value of the ‘lifetime’, then the route entry is deleted.

Routes may change due to the movement of a node (e.g. node X) within the path of the route. In such a case, the upstream neighbor of this node generates a ‘link failure notification message’ which notifies about the deletion of the part of the route and forwards this to its upstream neighbor. Upon reception of this message by a node, this is transmitted to the next upstream neighbor. The procedure continues until the source node is notified about the deletion of the route part caused by the movement of node X. Upon reception of the ‘link failure notification message’, the source node can reinitiate discovery of a route to the destination node.

**Dynamic Source Routing (DSR)**

DSR uses source routing, rather than hop-by-hop routing. Thus, in DSR every packet to be routed carries in its header the ordered list of network nodes that constitute the route over which the packet will be relayed. Thus, intermediate nodes do not need to maintain routing information as the contents of the packet itself are sufficient to route the packet. This fact eliminates the need for the periodic route advertisement and neighbor detection packets that are employed in other protocols. On the other hand, the overhead in DSR is larger, since each packet must contain the whole sequence of nodes comprising the route. Therefore, DSR will be most efficient in cases of networks of small diameter.

DSR comprises the processes of route discovery and route maintenance. A source node wishing to set up a connection to another node initiates the route discovery process by broadcasting a ROUTE\_REQUEST packet. This packet is received by neighboring nodes which in turn forward it to their own neighbors. A node forwards a ROUTE\_REQUEST message only if it has not yet been seen by this node and if the node’s address is not part of the route. The ROUTE\_REQUEST packet initiates a ROUTE\_REPLY upon reception of the ROUTE\_REQUEST packet either by the destination node or by an intermediate node that knows a route to the destination. Upon arrival of the ROUTE\_REQUEST message either to the destination or to an intermediate node that knows a route to the destination, the packet contains the sequence of nodes that constitute the route. This information is piggybacked on to the ROUTE\_REPLY message and consequently made available at the source node. DSR supports both symmetric and asymmetric links. Thus, the ROUTE\_REPLY message can be either carried over the same path with the original ROUTE\_REQUEST, or the destination

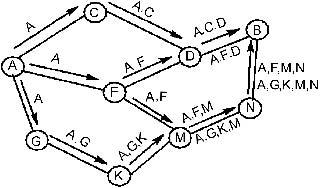


Figure: DSR route discovery

node might initiate its own route discovery towards the source node and piggyback the ROUTE\_REPLY message in its ROUTE\_REQUEST. Route discovery is shown schemati-cally in Figure 10.15 for an example network.

In order to limit the overhead of this control messaging, each node maintains a cache comprising routes that were either used by this node or overheard. As a result of route request by a certain node, all the possible routes that are learned are stored in the cache. Thus, a ROUTE\_REQUEST process may result in a number of routes being stored in the source node’s cache.

Route maintenance is initiated by the source node upon detection of a change in network topology that prevents its packets from reaching the destination node. In such a case the source node can either attempt to use alternative routes to the destination node (if such routes reside in the source’s cache) or reinitiate route discovery. Storing in the cache of alternative routes means that route discovery can be avoided when alternative routes for the broken one exist in the cache. Therefore route recovery in DSR can be faster than in other on-demand protocols.

Since route maintenance is initiated only upon link failure, DSR does not make use of periodic transmissions of routing information, resulting in less control signaling overhead and less power consumption at the mobile nodes.

**Associativity Based Routing (ABR)**

The fundamental objective of ABR is to find longer-lived routes for ad hoc mobile networks. This obviously results in fewer route reconstructions and thus higher throughput. ABR defines a new routing metric, called ‘degree of association’. This metric defines the level of association stability between neighboring nodes and is derived as follows: all nodes periodically generate and transmit beacons, in order to notify neighboring nodes of their existence. Beaconing intervals must be small enough to ensure accurate spatial and thus connectivity information. Whenever a node (e.g. A) receives such a beacon from a neighbor-ing node (e.g. B), it updates its associativity table by incrementing a counter which signifies the degree of association between this node and the beaconing neighbor. Associativity values are reset when the neighbors of a node or the node itself move out of range. Thus, for two neighboring nodes A and B, the value of the association counter described above defines the degree of association stability between the two nodes. High values of the associativity counter

for A and B indicate a low state of relative mobility, while a low value of the associativity counter may indicate a high state of node mobility.

ABR consists of three phases. These are described below:

* Route discovery. For purposes of route discovery, a node transmits a Broadcast Query (BQ) packet. This message contains the node’s address and the values of the associativity counter with its neighbors. Upon reception of a BQ message, a node erases its upstream neighbor value of the associativity counters and maintains only the associativity counter concerning itself and its upstream node. Then, it forwards the message to its downstream neighbors. A node does not forward a BQ request more than once. Thus, as a BQ packet reaches the destination node, it will contain the values of the associativity counters along the route from the source to the destination. Upon receiving a number of BQ packets (each one corresponding to a different path), the destination will posses information regarding the overall degree of association stability for each route and can thus select the best route. If more than two routes have the same association stability, then the one having the minimum number of hops is selected. Upon selection of a route by the destination, a REPLY packet is sent back to the source along the path specified by the route. As the REPLY packet traverses the path, the corresponding route is marked as active, while the alternative routes remain inactive. The above procedure is known as the BQ-REPLY process.
* Route reconstruction (RRC). Depending on which node (or nodes) along the route move, RRC consists of partial route discovery, invalid route erasure, valid route updates, and new route discovery. When the source node moves, a new BQ-REPLY process is initiated and the old route is deleted. When the destination node moves, then its immediate upstream neighbor erases its route and checks if the destination is still accessible by performing a localized query process (LQ[H], where H stands for the number of hops from the upstream node to the destination node). If the destination node receives the LQ packet, it selects the best partial route and issues a reply message. Otherwise, the upstream neighbor of the destination node concludes that the latter is out of range and the next upstream neighbor is instructed to perform the LQ process. This procedure continues until either a new route has been established or the process has backtracked more than half the number of hops that constituted the route from the source to the destination. In the latter case, the procedure is aborted and a new BQ-REPLY process starts at the source node.
* Route deletion (RD). An RD broadcast is initiated when a route is no longer valid. Upon reception of an RD packet, all nodes along the route delete the corresponding entries from their routing tables. RD messages are propagated by a full broadcast because the source node may not be aware of any route node changes that occurred during RRCs.

Signal Stability Routing (SSR)

SSR routes packets based on the signal strength between nodes and a node’s location stability. Thus, SSR selects those routes having the strongest connectivity. This fact aims at fewer route reconstructions and thus higher throughput.

SSR comprises two cooperative protocols. These are the Dynamic Routing Protocol (DRP) and the Static Routing Protocol (SRP). DRP maintains the Signal Stability Table (SST) and the Routing Table (RT). SST is used to store the signal strength of neighboring nodes. The

storage of these values in the SST is made possible by periodic link-layer beaconing of nodes in SSR. Based on the quality of the beacon signal, SST entries identify links as ‘weak’ or ‘strong’. All packet transmissions are monitored by the DRP before being passed to the node’s SRP which examines the packet in order to find out whether it is destined for this node or another one. In the first case, the packet is pushed up to higher protocol layers. In the second case the packet must be forwarded to its destination. Thus, the node searches in its RT for a route to the destination. If no route is found, then a route search process is initiated.

The fact that packets arriving over a weak channel are dropped at intermediate nodes means that route-search packets arriving at the destination have necessarily arrived on the path of strongest signal stability. Thus, the protocol routes packets over routes having the highest possible signal stability. However, under high BER conditions, few links may be classified as ‘strong’ In such cases, the route search process may not find a route to the destination. In such a case, the route search process initiator may chose to reinitiate the procedure indicating that weak links in the path of the route are acceptable.