**UNIT-4**

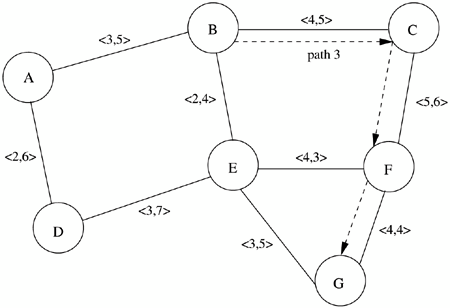
**QUALITY OF SERVICE IN AD HOC WIRELESS NETWORKS**

**4.1 INTRODUCTION**

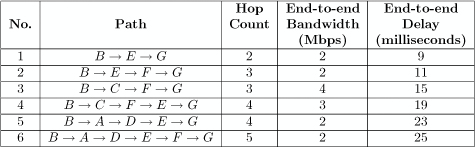
Quality of service (QoS) is the performance level of a service offered by the network to the user. The goal of QoS provisioning is to achieve a more deterministic network behavior, so that information carried by the network can be better delivered and network resources can be better utilized. A network or a service provider can offer different kinds of services to the users. Here, a service can be characterized by a set of measurable pre specified service requirements such as minimum bandwidth, maximum delay, maximum delay variance (jitter), and maximum packet loss rate. After accepting a service request from the user, the network has to ensure that the service requirements of the user's flow are met, as per the agreement, throughout the duration of the flow (a packet stream from the source to the destination). In other words, the network has to provide a set of service guarantees while transporting a flow.

After receiving a service request from the user, the first task is to find a suitable loop-free path from the source to the destination that will have the necessary resources available le to meet the QoS requirements of the desired service. This process is known as QoS routing. After finding a suitable path, a resource reservation protocol is employed to reserve necessary resources along that path. QoS guarantees can be provided only with appropriate resource reservation techniques. For example, consider the network shown in [Figure 4.1. T](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10fig01)he attributes of each link are shown in a tuple < *BW, D* >, where *BW* and *D* represent available bandwidth in Mbps and delay in milliseconds. Suppose a packet-flow from node*B* to node *G* requires a bandwidth guarantee of 4 Mbps. Throughout the chapter, the terms "node" and "station" are used interchangeably. QoS routing searches for a path that has sufficient bandwidth to meet the bandwidth requirement of the flow. Here, six paths are available between nodes *B* and *G* as shown in Table 4.1. QoS routing selects path 3 (*i.e., B* → *C* → *F* → G) because, out of the availab le paths, path 3 alone meets the bandwidth constraint of 4 Mbps for the flow. The end-to-end bandwidth of a path is equal to the bandwidth of the bottleneck link (*i.e.,* the link having minimum bandwidth among all the links of a path). The end-to-end delay of a path is equal to the sum of delays of all the links of a path. Clearly, path 3 is not optimal in terms of hop count and/or end-to-end delay parameters, while path 1 is optimal in terms of both hop count and end-to-end delay parameters. Hence, QoS routing has to select a suitable path that meets the QoS constraints specified in the service request made by the user. Delay includes transmission delay, propagation delay, and queuing delay.

**Figure 4.1. An example of QoS routing in ad hoc wireless network.**



**Table 4.1. Available paths from node *B* to node *G***



QoS provisioning often requires negotiation between host and network, call admission control, resource reservation, and priority scheduling of packets. QoS can be rendered in ad hoc wireless networks through several ways, namely, per flow, per link, or per node. In ad hoc wireless networks, the boundary between the service provider (network) and the user (host) is not defined clearly, thus making it essential to have better coordination among the hosts to achieve QoS. Characteristics of ad hoc wireless networks such as lack of central coordination, mobility of hosts, and limited availability of resources make QoS provisioning very challenging.

***4.1.1 Real-Time Traffic Support in Ad Hoc Wireless Networks***

Real-time applications require mechanisms that guarantee bounded delay and delay jitter. The end-to-end delay in packet delivery includes the queuing delay at the source and intermediate nodes, the processing time at the intermediate nodes, and the propagation duration over multiple hops from the source node to the destination node. Real-time applications can be classified as hard real-time applications and soft real-time applications. A hard real-time application requires strict QoS guarantees. Some of the hard real-time applications include nuclear reactor control systems, air traffic control systems, and missile control systems. In these applications, failure to meet the required delay constraints may lead to disastrous results. On the other hand, soft real-time applications can tolerate degradation in the guaranteed QoS to a certain extent. Some of the soft real-time applications are voice telephony, video-on-demand, and video conferencing. In these applications, the loss of data and variation in delay and delay jitter may degrade the service but do not produce hazardous results. Providing hard real-time guarantees in ad hoc wireless networks is extremely difficult due to reasons such as the unrestricted mobility of nodes, dynamically varying network topology, time-varying channel capacity, and the presence of hidden terminals. The research community is currently focusing on providing QoS support for applications that require soft real-time guarantees.

***4.1.2 QoS Parameters in Ad Hoc Wireless Networks***

As different applications have different requirements, the services required by them and the associated QoS parameters differ from application to application. For example, in case of multimedia applications, bandwidth, delay jitter, and delay are the key QoS parameters, whereas military applications have stringent security requirements. For applications such as emergency search-and-rescue operations, availability of the network is the key QoS parameter. Applications such as group communication in a conference hall require that the transmissions among nodes consume as little energy as possible. Hence, battery life is the key QoS parameter here. Unlike traditional wired networks, where the QoS parameters are mainly characterized by the requirements of multimed ia traffic, in ad hoc wireless networks the QoS requirements are more influenced by the resource constraints of the nodes. Some of the resource constraints are battery charge, processing power, and buffer space.

**4.2 ISSUES AND CHALLENGES IN PROVIDING QOS IN AD HOC WIRELESS NETWORKS**

Providing QoS support in ad hoc wireless networks is an active research area. Ad hoc wireless networks have certain unique characteristics that pose several difficulties in provisioning QoS. Some of the characteristics are dynamically varying network topology, lack of precise state information, lack of a central controller, error-prone shared radio channel, limited resource availability, hidden terminal problem, and insecure medium. A detailed discussion on how each of the above-mentioned characteristics affects QoS provisioning in ad hoc wireless networks is given below.

• **Dynamically varying network topology:** Since the nodes in an ad hoc wireless network do not have any restriction on mobility, the network topology changes dynamically. Hence, the admitted QoS sessions may suffer due to frequent path breaks, thereby requiring such sessions to be reestablished over new paths. The delay incurred in reestablishing a QoS session may cause some of the packets belonging to that session to miss their delay targets/deadlines, which is not acceptable for applications that have stringent QoS requirements.

• **Imprecise state information:** In most cases, the nodes in an ad hoc wireless network maintain both the link-specific state information and flow-specific state information. The link-specific state information includes bandwidth, delay, delay jitter, loss rate, error rate, stability, cost, and distance values for each link. The flow-specific information includes session ID, source address, destination address, and QoS requirements of the flow (such as maximum bandwidth requirement, minimum bandwidth requirement, maximum delay, and maximum delay jitter). The state information is inherently imprecise due to dynamic changes in network topology and channel characteristics. Hence, routing decisions may not be accurate, resulting in some of the real-time packets missing their deadlines.

• **Lack of central coordination:** Unlike wireless LANs and cellular networks, ad hoc wireless networks do not have central controllers to coordinate the activity of nodes. This further complicates QoS provisioning in ad hoc wireless networks.

• **Error-prone shared radio channel:** The radio channel is a broadcast medium by nature. During propagation through the wireless medium, the radio waves suffer from several impairments such as attenuation, multipath propagation.

• **Hidden terminal problem:** The hidden terminal problem is inherent in ad hoc wireless networks. This problem occurs when packets originating from two or more sender nodes, which are not within the direct transmission range of each other, collide at a common receiver node. It necessitates the retransmits ion of the packets, which may not be acceptable for flows that have stringent QoS requirements. The RTS/CTS control packet exchange mechanism, proposed in [[1]](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10biblio01entry01) and adopted later in the IEEE 802.11 standard, reduces the hidden terminal problem only to a certain extent.

• **Limited resource availability:** Resources such as bandwidth, battery life, storage space, and processing capability are limited in ad hoc wireless networks. Out of these, bandwidth and battery life are critical resources, the availability of which significantly affects the performance of the QoS provisioning mechanism. Hence, efficient resource management mechanisms are required for optimal utilization of these scarce resources.

• **Insecure medium:** Due to the broadcast nature of the wireless medium, communication through a wireless channel is highly insecure. Therefore, security is an important issue in ad hoc wireless networks, especially for military and tactical applications. Ad hoc wireless networks are susceptible to attacks such as eavesdropping, spoofing, denial of service, message distortion, and impersonation. Without sophisticated security mechanisms, it is very difficult to provide secure communication guarantees. Some of the design choices for providing QoS support are described below.

• **Hard state versus soft state resource reservation:** QoS resource reservation is one of the very important components of any QoS framework (a QoS framework is a complete system that provides required/promised services to each user or application). It is responsible for reserving resources at all intermediate nodes along the path from the source to the destination, as requested by the QoS session. QoS resource reservation mechanisms can be broadly classified into two categories: *hard state* and *soft state* reservation mechanisms. In hard state resource reservation schemes, resources are reserved at all intermed iate nodes along the path from the source to the destination throughout the duration of the QoS session. If such a path is broken due to network dynamics, these reserved resources have to be explicitly released by a deal location mechanism. Such a mechanism not only introduces additional control overhead, but may also fail to release resources completely in case a node previously belonging to the session becomes unreachable. Due to these problems, soft state resource reservation mechanisms, which maintain reservations only for small time intervals, are used. These reservations get refreshed if packets belonging to the same flow are received before the timeout period. The soft state reservation timeout period can be equal to packet inter- arrival time or a multiple of the packet inter-arrival time. If no data packets are received for the specified time interval, the resources are deallocated in a decentralized manner without incurring any additional control overhead. Thus no explicit teardown is required for a flow. The hard state schemes reserve resources explicitly and hence, at high network loads, the call blocking ratio will be high, whereas soft state schemes provide high call acceptance at a gracefully degraded fashion.

•**Stateful versus stateless approach:** In the stateful approach, each node maintains either *global state* information or only *local state* information, while in the case of a stateless approach, no such information is maintained at the nodes. State information includes both the topology information and the flow-specific information. If global state information is available, the source node can use a centralized routing algorithm to route packets to the destination. The performance of the routing protocol depends on the accuracy of the global state information maintained at the nodes. Significant control overhead is incurred in gathering and maintaining global state information. On the other hand, if mobile nodes maintain only local state information (which is more accurate), distributed routing algorithms can be used. Even though control overhead incurred in maintaining local state information is low, care must be taken to obtain loop-free routes. In the case of the stateless approach, neither flow- specific nor link-specific state information is maintained at the nodes. Though the stateless approach solves the scalability problem permanently and reduces the burden (storage and computation) on nodes, providing QoS guarantees becomes extremely difficult.

• **Hard QoS versus soft QoS approach:** The QoS provisioning approaches can be broadly classified into two categories: *hard QoS* and *soft QoS* approaches. If QoS requirements of a connection are guaranteed to be met for the whole duration of the session, the QoS approach is termed a hard QoS approach. If the QoS requirements are not guaranteed for the entire session, the QoS approach is termed a soft QoS approach. Keeping network dynamics of ad hoc wireless networks in mind, it is very difficult to provide hard QoS guarantees to user applications. Thus, QoS guarantees can be given only within certain statistical bounds. Almost all QoS approaches availab le in the literature provide only soft QoS guarantees.

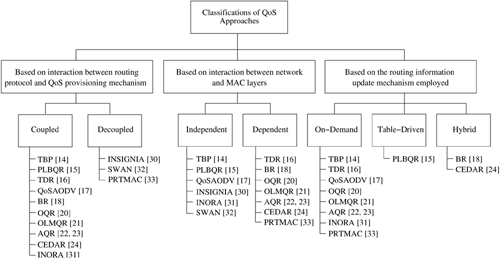
**4.3 CLASSIFICATIONS OF QOS SOLUTIONS**

The QoS solutions can be classified in two ways. One classificatio n is based on the QoS approach employed, while the other one classifies QoS solutions based on the layer at which they operate in the network protocol stack.

***4.3.1 Classifications of QoS Approaches***

As shown in [Figure 10.2, s](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10fig02)everal criteria are used for classifying QoS approaches. The QoS approaches can be classified based on the interaction between the routing protocol and the QoS provisioning mechanism, based on the interaction between the network and the MAC layers, or based on the routing information update mechanism. Based on the interaction between the routing protocol and the QoS provisioning mechanism, QoS approaches can be classified into two categories: *coupled* and *decoupled* QoS approaches. In the case of the coupled QoS approach, the routing protocol and the QoS provisioning mechanism closely interact with each other for delivering QoS guarantees. If the routing protocol changes, it may fail to ensure QoS guarantees. But in the case of the decoupled approach, the QoS provisioning mechanism does not depend on any specific routing protocol to ensure QoS guarantees.

**Figure 4.2. Classifications of QoS approaches.**

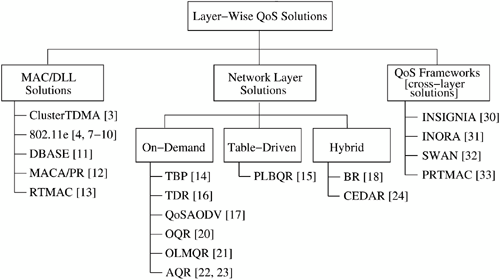


Similarly, based on the interaction between the routing protocol and the MAC protocol, QoS approaches can be classified into two categories: *independent* and *dependent* QoS approaches. In the independent QoS approach, the network layer is not dependent on the MAC layer for QoS provisioning. The dependent QoS approach requires the MAC layer to assist the routing protocol for QoS provisioning. Finally, based on the routing information update mechanism employed, QoS approaches can be classified into three categories, namely, *table-driven, on-demand,* and *hybrid* QoS approaches. In the table-driven approach, each node in the network maintains a routing table which aids in forwarding packets. In the on-demand approach, no such tables are maintained at the nodes, and hence the source node has to discover the route on the fly. The hybrid approach incorporates features of both the table-driven and the on-demand approaches.

***4.3.2 Layer-Wise Classification of Existing QoS Solutions***

The existing QoS solutions can also be classified based on which layer in the network protocol stack they operate in. Figure 4.3 gives a layer-wise classification of QoS solutions. The figure also shows some of the cross-layer QoS solutions proposed for ad hoc wireless networks. The following sections describe the various QoS solutions listed in Figure 4.3.

**Figure 4.3. Layer-wise classification of QoS solutions.**



**4.4 MAC LAYER SOLUTIONS**

The MAC protocol determines which node should transmit next on the broadcast channel when several nodes are competing for transmission on that channel. The existing MAC protocols for ad hoc wireless networks use channel sensing and random back-off schemes, making them suitable for best-effort data traffic. Real-time traffic (such as voice and video) requires bandwidth guarantees. Supporting real-time traffic in these networks is a very challenging task. In most cases, ad hoc wireless networks share a common radio channel operating in the ISM band or in military bands. The most widely deployed medium access technology is the IEEE 802.11 standard . The 802.11 standard has two modes of operation: a distributed coordination function (DCF) mode and a point coordination function (PCF) mode. The DCF mode provides best-effort service, while the PCF mode has been designed to provide real-time traffic support in infrastructure-based wireless network configurations. Due to lack of fixed infrastructure support, the PCF mode of operation is ruled out in ad hoc wireless networks. Currently, the IEEE 802.11 Task Group e (TGe) is enhancing the legacy 802.11 standard to support real-time traffic. The upcoming 802.11e standard has two other modes of operation, namely, enhanced DCF (EDCF) and hybrid coordination function (HCF) to support QoS in both infrastructure-based and infrastructure-less network configurations. These two modes of operation are discussed later in this section. In addition to these standardized MAC protocols, several other MAC protocols that provide QoS support for applications in ad hoc wireless networks have been proposed. Some of these protocols are described below. ISM refers to the industrial, scientific, and medical band. The frequencies in this band (from 2.4 GHz to 2.4835 GHz) are unlicensed.

***4.4.1 Cluster TDMA***

Gerla and Tsai proposed Cluster TDMA for supporting real-time traffic in ad hoc wireless networks. In bandwidth-constrained ad hoc wireless networks, the limited resources available need to be managed efficiently. To achieve this goal, a dynamic clustering scheme is used in Cluster TDMA. In this clustering approach, nodes are split into different groups. Each group has a cluster-head (elected by members of that group), which acts as a regional broadcast node and as a local coordinator to enhance the channel throughput. Every node within a cluster is one hop away from the cluster-head. The formation of clusters and selection of cluster-heads are done in a distributed manner. Clustering algorithms split the nodes into clusters so that they are interconnected and cover all the nodes. Three such algorithms used are lowest-ID algorithm, highest- degree (degree refers to the number of neighbors which are within transmission range of a node) algorithm, and least cluster change (LCC) algorithm. In the lowest-ID algorithm, a node becomes a cluster-head if it has the lowest ID among all its neighbors. In the highest-degree algorithm, a node with a degree greater than the degrees of all its neighbors becomes the cluster-head. In the LCC algorithm, cluster-head change occurs only if a change in network causes two cluster-heads to come into one cluster or one of the nodes moves out of the range of all the cluster-heads. The time division multiple access (TDMA) scheme is used within a cluster for controlling access to the channel. Further, it is possible for multiple sessions to share a given TDMA slot via code division multi le access (CDMA). Across clusters, either spatial reuse of the time-slots or different spreading codes can be used to reduce the effect of inter-cluster interference. A synchronous time division frame is defined to support TDMA access within a cluster and to exchange control information. Each synchronous time division frame is divided into slots. Slots and frames are synchronized throughout the network. A frame is split into a control phase and a data phase. In the control phase, control functions such as frame and slot synchronization, routing, clustering, power management, code assignment, and virtual circuit (VC) setup are done. The cluster-head does the reservation for the VC by assigning the slot(s) and code(s) to be used for that connection. The number of slots per frame to be assigned to a VC is determined by the bandwidth requirement of the VC. Each station broadcasts the routing information it has, the ID of its cluster-head, the power gain[3 l](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ftn.ch10fn03)ist (the power gain list consists of the power gain values corresponding to each of the single-hop neighbors of the node concerned) it maintains, reservation status of the slots present in its data phase, and ACKs for frames that are received in the last data phase. Upon receiving this information, a node updates its routing table, calculates power gains for its neighbors, updates the power gain matrix, selects its cluster-head, records the slot reservation status of its neighbors, obtains ACKs for frames that are transmitted in the last data phase, and reserves slot(s). In each cluster, the corresponding cluster-head maintains a power gain matrix. The power gain matrix contains the power gain lists of all the nodes that belong to a particular cluster. It is useful for controlling the transmission power and the code division within a cluster. Power gain is the power propagation loss from the transmitter to the receiver. The data phase supports both real-time and best-effort traffic. Based on the bandwidth requirement of the real-time session, a VC is set up by allocating sufficient number of slots in the data phase. The remaining data slots (*i.e.,* free slots) can be used by the best-effort traffic using the slotted-ALOHA scheme. For each node, a predefined slot is assigned in the control phase to broadcast its control information. The control information is transmitted over a common code throughout the network. At the end of the control phase, each node would have learned from the information broadcast by the cluster-head, the slot reservation status of the data phase and the power gain lists of all its neighbors. This information helps a node to schedule free slots, verify the failure of reserved slots, and drop expired real-time packets. A fast reservation scheme is used in which a reservation is made when the first packet is transmitted, and the same slots in the subsequent frames can be used for the same connection. If the reserved slots remain idle for a certain timeout period, then they are released.

***4.4.2 IEEE 802.11e***

In this section, the IEEE 802.11 MAC protocol is first described. Then, the recently proposed mechanisms for QoS support, namely, enhanced distributed coordination function (EDCF) and hybrid coordination function (HCF), defined in the IEEE 802.11e draft, are discussed.

**IEEE 802.11 MAC Protocol**

It supports two modes of operation, namely, distributed coordination function (DCF) and point coordination function (PCF). The DCF mode does not use any kind of centralized control, while the PCF mode requires an access point (AP, *i.e.,* central controller) to coordinate the activity of all nodes in its coverage area. All implementations of the 802.11 standard for WLANs must provide the DCF mode of operation, while the PCF mode of operation is optional. The time interval between the transmission of two consecutive frames is called the inter-frame space (IFS). There are four IFSs defined in the IEEE 802.11 standard, namely, short IFS (SIFS), PCF IFS(PIFS), DCF IFS (DIFS), and extended IFS (EIFS). The relationship among them is as follows:



Distributed Coordination Function In the DCF mode, all stations are allowed to contend for the shared medium simultaneously. CSMA/CA mechanism and random back-off scheme are used to reduce frame collisions. Each uncast frame is acknowledged immed iately after being received. If the acknowledgment is not received within the timeout period, the data frame is retransmitted. Broadcast frames do not require acknowledgments from the receiving stations. If a station *A* wants to transmit data to station *B*, station *A* listens to the channel. If the channel is busy, it waits until the channel becomes idle. After detecting the idle channel, station *A* further waits for a DIFS period and invokes a back-off procedure. The back-off time is given by



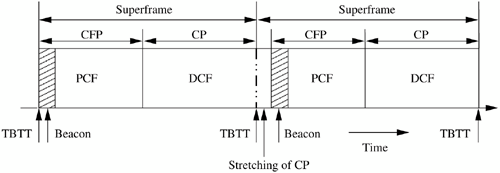
where *slot time* includes the time needed for a station to detect a frame, the propagation delay, the time needed to switch from the receiving state to the transmitting state, and the time to signal to the MAC layer the state of the channel. The function *rand*(0, *CW*) returns a pseudo-random integer from a uniform distribution over an interval [0, *CW*]. The current value of the

contention window (*CW*) plays an important role in determining the back-off period of the station. The initial value of *CW* is *CW min* . If a collision occurs, the value of *CW* is doubled. As the number of collisions increases, the value of *CW* is increased exponentially in order to reduce the chance of collision occurrence. The maximum value of *CW* is *CW max* . After detecting the channel as being idle for a DIFS period, station *A* starts decrementing the back-off counter. If it senses the channel as busy during this count-down process, it suspends the back-off counter till it again detects the channel as being idle for a DIFS period. Station *A* then continues the count- down process, where it suspended the back-off counter. Once the back-off counter reaches zero, station *A* transmits a request-to-send (RTS) frame and waits for a clear-to-send (CTS) frame from the receiver *B.* If other stations do not cause any interference, station *B* acknowledges the RTS frame by sending a CTS frame. Upon receiving the CTS frame, station *A* transmits its data frame, the reception of which is acknowledged by receiver *B* by sending an ACK frame. In the above scenario, if another station *C* apart from station *A* also senses the channel as being idle (*i.e.,* stations *A* and *C* sense the channel as being idle and the back-off counters set by them expire at the same time) and transmits an RTS frame, a collision occurs and both the stations initiate back-off procedures. If the size of the MAC frame, that is, MAC service data unit (MSDU), [i](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ftn.ch10fn04)s greater than the fragmentation threshold, it is fragmented into smaller frames, that is, MAC protocol data units (MPDUs), [b](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ftn.ch10fn05)efore transmission, and each MPDU has to be acknowledged separately. Once an MSDU is transmitted successfully, *CW* is reset to *CW min* . The RTS/CTS control frame exchange helps in reducing the hidden terminal problem inherent in CSMA-based ad hoc wireless networks. MSDU is the information that is delivered as a unit between MAC service access points. MPDU is the unit of data exchanged between two peer MAC entities using the services of the physical layer.

**Point Coordination Function**

The IEEE 802.11 standard incorporates an optional access method known as PCF to let stations have priority access to the wireless medium. This access method uses a point coordinator (PC), which operates at an AP. Hence PCF is usable only in infrastructure-based network configurations. A station which requires the PCF mode of operation sends an association message to the PC to register in its polling list and gets an association identifier (AID). The PC polls the stations registered in its polling list in ascending order of AIDs to allow them contention-free access to the medium. The role of the PC is to determine which station should gain access to the channel. The stations requesting the PCF mode of operation get associated with the PC during the contention period (CP). With PCF, the channel access alternates between the contention-free period (CFP) and the contention period (CP) for the PCF and DCF modes of operation, respectively. A CFP and the following CP form a super frame. The PC generates a beacon frame at regular beacon frame intervals called target beacon transmission time (TBTT). The value of TBTT is announced in the beacon frame. Each super frame starts with a beacon frame, which is used to maintain synchronization among local timers in the stations and to deliver protocol- related parameters. The PC uses contention-free poll (*CF-Poll*) packets to ask stations to transmit their frames. A station that is able to respond to*CF-Poll* frames is said to be *CF-Pollable.* It is optional for a *CF-Pollable* station to respond to a *CF-Poll*frame received from the PC. If the PC receives no response from the polled station for a PIFS period, it polls the next station in the polling list (in case the remaining duration of CFP is long enough for at least one CFP transmission) or ends the CFP by transmitting *CF-End* control frame. The PC and the *CF-Pollable* stations do not use the RTS/CTS control frame exchange in the CFP. [Figure 4.4 s](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10fig04)hows the operation of the network in the combined PCF and DCF modes. The channel access switches alternately between the PCF mode and the DCF mode, but the CFP may shrink due to stretching when DCF takes more time than expected. This happens when an MSDU is fragmented into several MPDUs, hence giving priority to these fragments over the PCF mode of operation.

**Figure 4.4. PCF and DCF frame sharing.**



PCF has certain shortcomings which make it unsuitable for supporting real-time traffic.. At TBTT, the PC has to sense the medium idle for at least PIFS before transmitting the beacon frame. If the medium is busy around TBTT, the beacon is delayed, thereby delaying the transmission of real-time traffic that has to be delivered in the following CFP. Further, polled stations' transmission durations are unknown to the PC. The MAC frame (*i.e.,* MSDU) of the polled station may have to be fragmented and may be of arbitrary length. Further, the transmission time of an MSDU is not under the control of the PC because of different modulation and coding schemes specified in the IEEE 802.11 standard. PCF is not scalable to support real-time traffic for a large number of users. Due to these reasons, several mechanisms have been proposed to enhance the IEEE 802.11 standard to provide QoS support. The QoS mechanisms that are proposed as part of the IEEE 802.11e standard are described below.

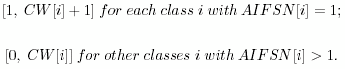
**QoS Support Mechanisms of IEEE 802.11e:**

The IEEE 802.11 WLAN standard supports only best-effort service. The IEEE 802.11 Task Group e (TGe) has been set up to enhance the current. 802.11 MAC protocol so that it is able to support multimedia applications. The TGe has chosen the virtual DCF (VDCF) proposal as the enhanced DCF (EDCF) access mechanism. EDCF supports real-time traffic by providing differentiated DCFaccess to the wireless medium. The TGe has also specified a hybrid coordination function (HCF) that combines EDCF with the features of PCF to simplify the QoS provisioning. HCF operates during both the CFP and the CP. Enhanced Distributed Coordination Function Enhanced distributed coordination function (EDCF) provides differentiated and distributed access to the wireless medium. Each frame from the higher layer carries its user priority (UP). After receiving each frame, the MAC layer maps it into an access category (AC). Each AC has a different priority of access to the wireless medium. One or more UPs can be assigned to each AC. EDCF channel access has up to eight ACs , to support UPs. EDCF supports eight UPs. Similar to the DCF, each AC has a set of access parameters, such as *CW min , CW max , AIFS*, and transmission opportunity (TXOP) limit, which would be described later in this section. Hence, each AC is an enhanced variant of the DCF. Flows that fall under the same AC are effectively given identical priority to access the channel. A station accesses the channel based on the AC of the frame to be transmitted. An access point that provides QoS is called QoS access point (QAP). Each QAP will provide at least four ACs. Each station contends for transmission opportunities (TXOPs) using a set of EDCF channel access parameters that are unique to the AC of the packet to be transmitted. The TXOP is defined as an interval of time during which a station has the right to initiate transmissions. It is characterized by a starting time and a maximum duration called TXOP Limit. Depending on the duration of TXOP, a station may transmit one or more MSDUs. Priority of an AC refers to the

lowest UP assigned to that AC. During CP, each AC (of priority *i*) of the station contends for a TXOP and independently starts a back-off counter after detecting the channel being idle for an arbitration inter-frame space (*AIFS*[*i*]) as specified in [[10]](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10biblio01entry10). *AIFS*[*i*] is set as given below.



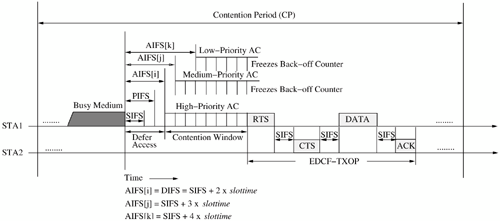
where *AIFSN*[*i*] is the *AIFS slot count* (*i.e.,* the number of time-slots a station has to sense the channel as idle before initiating the back-off process) for priority class *i* and takes values greater than zero. For high-priority classes, low *AIFSN* values are assigned to give higher priorities for them. After waiting for *AIFS*[*i*], each back-off counter is set to a random integer drawn from the range:



The reason for having a different range for classes with *AIFSN*[*i*] = 1 is to avoid transmissions initiated by stations that are operating in the EDCF mode from colliding with the hybrid coordinator's (HC, which is explained later in this section) poll packets. The HC operates at QAP and controls QoS basic service set (QBSS) operation under the HCF. [Figure 4.5 i](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10fig05)llustrates the relationship between SIFS, PIFS, DIFS, and various AIFS values. As in legacy DCF, if a station detects the channel to be busy before the back-off counter reaches zero, the back-off counter is suspended. The station has to wait for the channel to become idle again for an AIFS period, before continuing to decrement the counter. In this figure, it is assumed that station *ST A*1 has traffic that belongs to three different ACs. The back-off counter of the highest priority AC expires first, which causes the corresponding AC to seize an EDCF-TXOP for initiating data transmission. The other ACs suspend their back-off counters and wait for the channel to become idle again. When the back-off counter of a particular AC reaches zero, the corresponding station initiates a TXOP and transmits frame(s) that have the highest priority. TXOPs are allocated via contention (EDCF-TXOP) or granted through HCF (polled-TXOP) . The duration of EDCF-TXOP is limited by a QBSS-wide TXOP Limit transmitted in beacons by the HC, while during the CFP the starting time and maximum duration of each polled-TXOP is specified in the corresponding *CF-Poll* frame by the HC. If the back-off counters of two or more ACs in a single station reach zero at the same time, a scheduler inside the station avoids the *virtual collision* by granting the TXOP to the highest priority AC, while low-priority

ACs behave as if there was an external collision on the wireless medium.

**Figure 4.5. An example of EDCF access mechanism.**



**Hybrid Coordination Function**

The hybrid coordination function (HCF) combines features of DCF and PCF to provide the capability of selectively handling MAC service data units (MS-DUs), in a manner that has upward compatibility with both DCF and PCF. It uses a common set of frame exchange sequences during both the CP and the CFP. The HCF is usable only in infrastructure-based BSSs that provide QoS, that is, QBSSs. The HCF uses a QoS-aware point coordinator, called HC, which is typically collocated with aQAP. The HC implements the frame exchange sequences and the MSDU handling rules defined inHCF, operating during both the CP and the CFP. It allocates TXOPs to stations and initiates controlled contention periods for the stations to send reservation requests. When the HC needs access to the wireless medium, it senses the medium. If the medium remains idle for a PIFS period, it initiates MSDU deliveries. The HC can start contention-free controlled access periods (CAPs) at any time during a CP, after the medium is determined to be idle for at least one PIFS period.

A CAP may include one or more TXOPs. During the CAP, the HC may transmit frames and issue polls to stations which grant them TXOPs. At the end of the TXOP or when the station has no more frames to transmit, it explicitly hands over control of the medium back to the HC. [Figure 4.6 s](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10fig06)hows an example of a super frame divided into CFP, CP, and three CAP intervals. During CP, each TXOP begins either when the medium is determined to be availab le under the EDCF rules (EDCF-TXOP) or when the station receives a QoS *CF-Poll* frame from the HC (polled-TXOP).

**Figure 4.6. Division of time into CFP, CP, and CAP intervals.**

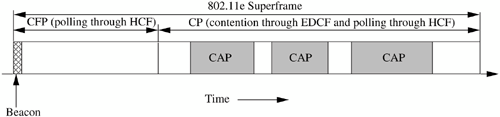
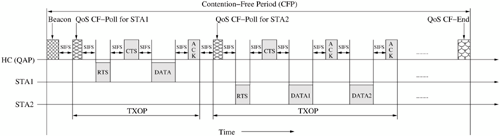


Figure 4.7 illustrates CFP in the HCF mode of operation. During CFP, the HC grants TXOPs to stations by sending QoS *CF-Poll* frames. The polled station can transmit one or more MSDUs in the allocated TXOP. If the size of an MSDU is too large, it can be divided into two or more fragments and transmitted sequentially with SIFS waiting periods in between them. These fragments have to be acknowledged individually. The CFP ends after the time announced in the beacon frame or by a *CF-End* frame from the HC.

**Figure 4.7. An example of HCF access mechanism.**



***4.4.3 DBASE***

The distributed bandwidth allocation/sharing/extension (DBASE) protocol supports multimedia traffic [both variable bit rate (VBR) and constant bit rate (CBR)] over ad hoc WLANs. In an ad hoc WLAN, there is no fixed infrastructure (*i.e.,* AP) to coordinate the activity of individual stations. The stations are part of a single-hop wireless network and contend for the broadcast channel in a distributed manner. For real-time traffic (*rt-traffic*), a contention- based process is used in order to gain access to the channel. Once a station gains channel access, a reservation-based process is used to transmit the subsequent frames. The non-real-time stations (*nrt-*stations) regulate their accesses to the channel according to the standard CSMA/CA protocol used in 802.11 DCF. The DBASE protocol permits real-time stations (*rt*-stations) to acquire excess bandwidth on demand. It is still compliant with the IEEE 802.11 standard. Like the IEEE 802.11 standard, the DBASE protocol divides the frames into three priority classes. Frames belonging to different priority classes have to wait for different IFSs before they are transmitted. Stations have to wait for a minimum of PIFS before transmitting *rt*-frames such as reservation frame (RF) and request-to-send (RTS). The *nrt-*frames have the lowest priority, and hence stations have to wait for DIFS before transmitting such frames. The Access Procedure for Non Real-Time Stations The channel access method for *nrt*-stations is based on conventional DCF. An *nrt-*station with data traffic has to keep sensing the channel for an additional random time called data back-off time (DBT) after detecting the channel as being idle for a DIFS period. The DBT is given by



The function *rand*(*a, b*) returns a pseudo-random integer from a uniform distribution over an interval [*a*, *b*], where *b* grows exponentially for each retransmission attempt, and the range of *b* is between *b min* and *b max* . DBASE adopts the contention window parameters from

the IEEE 802.11 DSSS specification. If the channel is idle, the DBT counter is decremented till it reaches zero, but it is frozen while the channel becomes busy. Once the DBT counter reaches zero, the *nrt*-station transmits its *nrt*-frame. The destination sends an ACK to the source after SIFS period after receiving the *nrt-* frame correctly from the source.

The Access Procedure for Real-Time Stations Each *rt*-station maintains a virtual reservation table (RSVT). In this virtual table, the information regarding all *rt*-stations that have successfully reserved the required bandwidth is recorded. Before initiating an *rt*-session, the *rt*-station sends an RTS in order to reserve the required bandwidth. Before transmitting the RTS, a corresponding entry is made in the RSVT of the node. Every station that hears this RTS packet also makes a corresponding entry in its RSVT. After recording into the RSVT successfully, an *rt*-station need not contend for the channel any more during its whole session. Bandwidth Reservation One of the *rt*-stations takes the responsibility of initiating the contention-free period (CFP) periodically. Such an *rt*-station is designated as CFP generator (CFPG). The CFP is utilized by the active *rt*-stations present in the network to transmit their *rt*-frames. The CFPG issues a reservation frame (RF) periodically and has the right to send its *rt*-frame first in the CFP. The maximum delay between any two consecutive RFs is *D max* , where *D max* is the minimum of maximum delay bounds among all active *rt*-connections. The RF is a broadcast

frame that announces the beginning of the CFP. The RF contains the information about the number of active *rt*-stations and the information about all*rt*-stations recorded in the RSVT of the CFPG. Assume that at time *t* an *rt*-station wants to transmit data. Then it monitors the channel for detecting the RF during the interval (*t*, *t* + *D max* ). If the *rt-*station detects the RF, it waits until the CFP finishes. After the CFP finishes, the *rt*- station keeps sensing the channel for a period of real-time back-off time (RBT) after detecting the channel as being idle for a PIFS period. The RBT of an *rt*- station is given by

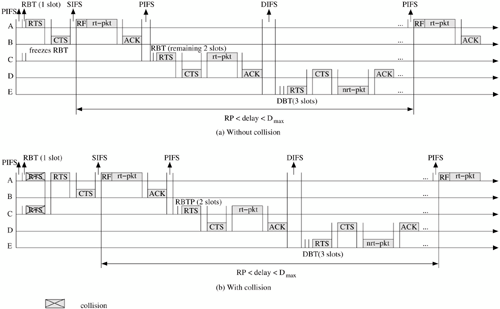


where *rand*(*c, d*) returns a pseudo-random integer from a uniform distribution over an interval [*c, d*]. The values of *c* and *d* are set to 0 and 3, respectively. If the channel is idle, the RBT counter is decremented till it reaches zero, but it is frozen while the medium is sensed busy. Once the RBT counter reaches zero, the *rt*-station contends for its reservation by sending an RTS packet. If no collision occurs, it updates its tables and transmits its first *rt*-frame. If a collision occurs, the *P*-persistent scheme is used to resolve the contention. The *rt*-station involved in collision retransmits theRTS in the next time-slot (*i.e., slottime*) with a probability *P*. With probability (1 - *P*), it defers for at least one time-slot and recalculates the RBT using the following equation:



where RBTP is the recalculated RBT for the *P*-persistent scheme. If an RF is not received during the interval (*t, t* + *D max* ), it means that there are no active *rt*-stations. If the channel is still idle in the interval (*t* + *D max* + *δ ,t* + *D max* + *δ* + *PIFS*) and no RF is detected, the *rt-*station that wants to transmit data at time instant *t* will execute the back-off scheme. Here *δ* represents the remaining transmitting time of the current frame at the time instant *t* + *D max* . During the back-off process, the *rt*-station should keep monitoring the channel to check whether any *rt*-station has started acting as the CFP generator. If RBT reaches zero, the *rt*-station sends an RTS frame to the receiver. If no collision occurs, it gets CTS from the receiver and plays the role of CFPG in the network. If a collision occurs, the *P*-persistent scheme as mentioned above is used to decide when the stations are to transmit again. The bandwidth reservation scheme is illustrated in Figure 4.8. Figure 4.8 [(a) d](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10fig08)epicts a case in which no collision occurs, while Figure 4.8 (b) shows a scenario in which a collision occurs. In Figure 4.8 (a), stations *A* and *C* have *rt*- frames for transmission to stations *B* and *D,* respectively. Besides these, station *E* has *nrt*-frames to be transmitted to station *D.* After listening to the channel for *D max* time period in order to detect the presence of an RF, stations *A* and *C* conclude that no CFPG exists in the network. Then, if they find the channel as being idle for a PIFS period, they initiate their back-off timers. In this case, assume that *RBT A* is one slot and *RBT C* is three slots. During the back- off process, once the channel becomes busy, the back-off timer of station *C* is paused as shown in Figure 4.8 (a). It is restarted from the same value once the channel becomes idle again. After *RBT A* counts down to zero, station *A* seizes the channel and sends an RTS. When station *A* starts transmitting, station *C* pauses its back-off counter. If no collision occurs, station *A* receives a CTS within SIFS time duration. Then station *A* records its reservation information into the RSVT and becomes the CFPG. Since station *A* is currently playing the role of CFPG, it transmits an RF before transmitting its first *rt*- frame. Once station *A* completes its transmission, station *C* continues its back- off process. When *RBT C* counts down to zero, station *C* reserves bandwidth by adding a corresponding entry into the RSVT and transmits its first *rt*-frame. When station *E* detects the channel as being idle for DIFS, it implies that no other *rt*-station wants to transmit currently, and hence station *E* sends its RTS as soon as *DBT E* counts down to zero. By the end of a contention period whose length is limited by a parameter *RP max* (maximum repetition period), bandwidth would be reserved for the *rt*-stations, and thereafter they need not exchange RTS/CTS control frames before transmitting their *rt*-frames. The delay between two RFs varies from real-time period (RP) to *D max* , where RP is the sum of the CFP (*rt-*stations reserved period) and the CP for new *rt*-stations.

**Figure 4..8. An example of new *rt*-stations joining the network.**



In Figure 4.8 (b), assume that both station *A* and station *C* generate RBT as one slot. After waiting for one time-slot, both transmit their RTS frames, which results in a collision. Then the *P*-persistent scheme is applied. Assume that

station *A* gets access to the channel during the next slot itself, but station *C* does not. Then, station *A* will retransmit its RTS in the following slot, while station *C* initiates a new back-off time *RBTP C* . If no collision occurs, station *A* gets a CTS within SIFS, and sends out an RF and its *rt*-frame. When *RBTP C* counts down to zero, station *C* seizes the channel to send an RTS. If any collision occurs, the *rt*-station uses the *P*-persistent scheme to resolve the collision. The collision resolution process is restricted from crossing the *RP max* boundary. The MAC layer solutions such as MACA/PR and RTMAC provide real-time traffic support in asynchronous ad hoc wireless networks.

**4.5 NETWORK LAYER SOLUTIONS**

The bandwidth reservation and real-time traffic support capability of MAC protocols can ensure reservation at the link level only, hence the network layer support for ensuring end-to-end resource negotiation, reservation, and reconfiguration is very essential. This section describes the existing network layer solutions that support QoS provisioning.

***4.5.1 QoS Routing Protocols***

QoS routing protocols search for routes with sufficient resources in order to satisfy the QoS requirements of a flow. The information regarding the availability of resources is managed by a resource management module which assists the QoS routing protocol in its search for QoS feasible paths. The QoS routing protocol should find paths that consume minimum resources. The QoS metrics can be classified as additive metrics, concave metrics, and multiplicative metrics. An additive metric *Am* is defined as , where *L i* (*m*) is the value of metric *m* over link *L i* and *L i* × P. The hop length of path P is *h*. A concave metric represents the minimum value over a path P and is formally defined as *C m* = *min*(*L i* (*m*)), *L i* (*m*) ε P. A multip licative metric represents the product of QoS metric values and is defined as

,

*L i* (*m*) ε P. To find a QoS feasible path for a concave metric, the available resource on each link should be at least equal to the required value of the metric. Bandwidth is a concave metric, while cost, delay, and delay jitter are additive metrics. The reliab ility or availab ility of a link, based on some criteria such as link-break-p robab ility, is a multip licative metric. Finding an optimal path with multip le constraints may be an NP-complete problem if it involves two or more additive metrics. For example, finding a delay-constrained least-cost path is an NP-complete Problem. To assist QoS routing, the topology information can be maintained at the nodes of ad hoc wireless networks. The topology information needs to be refreshed frequently by sending link state update messages, which consume precious network resources such as bandwidth and battery power. Otherwise, the dynamically varying network topology may cause the topology informatio n to become imprecise. This trade-off affects the performance of the QoS routing protocol. As path breaks occur frequently in ad hoc wireless networks, compared to wired networks where a link goes down very rarely, the path satisfying the QoS requirements needs to be recomputed every time the current path gets broken. The QoS routing protocol should respond quickly in case of path breaks and recompute the broken path or bypass the broken link without degrading the level of QoS. In the literature, numerous routing protocols have been proposed for finding QoS paths. In the following sections, some of these QoS routing protocols are described.

***4.5.2 Ticket-Based QoS Routing Protocol***

Ticket-based QoS routing is a distributed QoS routing protocol for ad hoc wireless networks. This protocol has the following features:

• It can tolerate imprecise state information during QoS route computation and exhibits good performance even when the degree of imprecision is high.

• It probes multip le paths in parallel for finding a QoS feasible path. This increases the chance of finding such a path. The number of multip le paths searched is limited by the number of tickets issued in the probe packet by the source node. State information maintained at interned nodes is used for more accurate route probing. An intelligent hop-by-hop selection mechanism is used for finding feasible paths efficiently.

• The optimality of a path among several feasible paths is explored. A low-cost path that uses minimum resources is preferred when multiple feasible paths are available.

• A primary-bac kup-based fault-tolerant technique is used to reduce service disruption during path breaks that occur quite frequently in ad hoc wireless networks.

**Protocol Overview**

The basic idea of the ticket-based probing protocol is that the source node issues a certain number of tickets and sends these tickets in probe packets for finding a QoS feasible path. Each probe packet carries one or more tickets. Each ticket corresponds to one instance of the probe. For example, when the source node issues three tickets, it means that a maximum of three paths can be probed in parallel. The number of tickets generated is based on the precision of state information available at the source node and the QoS requirements of the connection request. If the available state information on is not precise or if the QoS requirements are very stringent, more tickets are issued in order to improve the chances of finding a feasible path. If the QoS requirements are not stringent and can be met easily, fewer tickets are issued in order to reduce the level of search, which in turn reduces the control overhead. There exists a trade-off here between the performance of the QoS routing protocol and the control overhead. The state information, at the source node, about intermediate nodes is useful in finding a much better QoS path, even if such information is not precise. The state information maintained at each node is comprised of estimations of end-to- end delay and available path bandwidth for every other node present in the network. When an intermed iate node receives a probe packet, it is either split to explore more than one path or is forwarded to just one neighbor node based on the state information available at that intermed iate node. Based on the idea of ticket-based probing, two heuristic algorithms are proposed, one for delay-constrained QoS routing, and the other for bandwidth- constrained QoS routing. In delay-constrained QoS routing, each probe accumulates the delay of the path it has traversed so far. In other words, if an intermed iate node *A* receives a probe packet (PKT) from a neighbor node *B,* node *A* updates the delay field in PKT by adding delay value of the link between nodes *B* and *A*. Then node *A* determines the list of candidate neighbors to which it has to send probe packets. It distributes tickets present in PKT among these new probe packets and then forwards these probe packets to the respective candidate neighbors. If multiple probe packets arrive at the destination node (with each carrying the list of intermed- iate nodes along its path), node *A* selects the path with least cost as the primary path and the other paths as the backup paths which will be used when the primary path is broken due to the mobility of intermed iate nodes.

**Optimizing Cost of a Feasible Path**

This protocol searches for the lowest cost path among the feasible paths. This is done during the QoS path probing. The source node issues two types of tickets, yellow tickets and green tickets, and sends them along with probe packets. Yellow tickets prefer paths that satisfy the requirement of a probe in terms of QoS metrics. For example, in delay-constrained QoS routing, yellow tickets are used to search for paths that have least delay, such that the end-to-end delay requirement is met. If the delay requirement is very large and can be met easily, only one yellow ticket is issued. If the delay requirement is too small to be met, then the source node does not issue any yellow ticket and rejects the connection request. Otherwise, more than one yellow ticket is issued to search multiple paths for finding a feasible QoS path. Green tickets are used to search for QoS paths with low costs. Similar to the manner in which the source node determines the number of yellow tickets, it also determines the number of green tickets to be issued on the basis of the delay requirement of the connection request. The distribution of yellow and green tickets (by an intermed iate node to its candidate neighbors) is based on the delay and cost requirements of the connection request, respectively. The concept behind two types of tickets is to use the more aggressive green tickets to find a least cost feasible path, and use yellow tickets as a backup to maximize the probability of finding a feasible path.

**Path Rerouting and Path Maintenance**

This protocol suggests a primary-backup-based, fault-tolerant technique to cope up with the network dynamics. To tolerate faults, a multi-level redundancy scheme is proposed. For the highest level of redundancy, multiple paths (preferably disjoint) are probed and data is routed independently on all paths. The destination node selects the first data copy and discards other copies which arrive later. Another level of redundancy which requires less resources has also been proposed. Here, one path is selected as the primary path and other paths (having resources reserved) act as backup paths. The third type of redundancy incurs even less control overhead and consumes very few resources. Here, backup paths are available along with the primary path, but resources are not reserved in these backup paths. During path maintenance, in order to eliminate the broken link, the call is rerouted over a backup path which has enough resources to satisfy the QoS requirements of the call. In the case of the third type of redundancy, since no resource reservation has been done along backup paths, during path breaks it is extremely difficult to find a backup that has enough resources to satisfy the QoS requirements of the call.

**Advantages and Disadvantages**

The objective of ticket-based probing is to improve the average call acceptance ratio (ACAR) of ad hoc wireless networks. ACAR is the ratio of the number of calls accepted to the number of calls received by the network. The protocol adapts dynamically to the requirements of the application and the degree of imprecision of state information maintained. It offers a trade-off between control overhead incurred in finding a feasible path and the cost of a feasible path. As the maximum number of probes in the network is equal to the number of tickets issued, the control overhead is bound by the number of tickets. The performance of the protocol depends on the ticket-issuing mechanism at the source node and the ticket-splitting procedure at the intermediate nodes. The protocol assumes that each node has global state information, but maintaining such information incurs huge control overhead in the already bandwidth-constrained ad hoc wireless networks. The proposed heuristic algorithms, which are based on an imprecise state information model, may fail in finding a feasible path in the extreme cases where the topology changes very rapidly. In delay-constrained QoS routing, the queuing delay and the processing delay at the intermediate nodes are not taken into consideration while measuring the delay experienced so far by the probe packet. This may cause some data packets to miss their deadlines. The routing algorithm works well only when the average lifetime of an established path is much longer than the average rerouting time. During the rerouting process, if QoS requirements are not met, data packets are transmitted as best-effort packets. This may not be acceptable for applications that have stringent QoS requirements.

**4.5.3 Predictive Location-Based QoS Routing Protocol**

The predictive location-based QoS routing protocol (PLBQR) is based on the prediction of the location of nodes in ad hoc wireless networks. The prediction scheme overcomes to some extent the problem arising due to the presence of stale routing information. No resources are reserved along the path from the source to the destination, but QoS-aware admission control is performed. The network does its best to support the QoS requirements of the connection as specified by the application. The QoS routing protocol takes the help of an update protocol and location and delay prediction schemes. The update protocol aids each node in broadcasting its geographic location and resource information to its neighbors. Using the update messages received from the neighbors, each node updates its own view of the network topology. The update protocol has two types of update messages, namely, *Type 1 update*and *Type 2 update.* Each node generates a Type 1 update message periodically. A Type 2 update message is generated when there is a considerable change in the node's velocity or direction of motion. From its recent update messages, each node can calculate an expected geographical location where it should be located at a particular instant and then periodically checks if it has deviated by a distance greater than *δ* from this expected location. If it has deviated, a Type 2 update message is generated.

**Location and Delay Predictions**

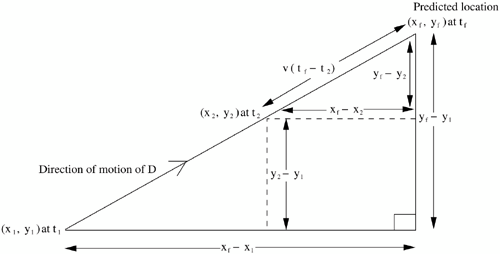
In establishing a connection to the destination *D,* the source *S* first has to predict the geographic location of node *D* and the intermediate nodes, at the instant when the first packet reaches the respective nodes. Hence, this step involves location prediction as well as propagation delay prediction. The location prediction is used to predict the geographic location of the node at a particular instant *tf* in the future when the packet reaches that node. The propagation delay prediction is used to estimate the value of *tf* used in the above location prediction. These predictions are performed based on the previous update messages received from the respective nodes.

**Location Prediction**

Let (*x*1 , *y*1 ) at *t*1 and (*x*2 , *y*2 ) at *t2* (*t*2 > *t*1 ) be the latest two updates from the destination *D* to the source node *S*. Assume that the second update message also indicates *v*, which is the velocity of *D* at (*x*2 , *y*2 ). Assume that node *S* wants to predict the location (*xf , yf* ) of node *D* at some instant *tf* in the future. This situation is depicted in Figure 4.9. The value of *t*f has to be estimated first using the delay prediction scheme, which will be explained later in this section. From Figure 4.9, using similarity of triangles, the following equation is obtained:



**Figure 4.9. Prediction of location at a future time by node S using the last two updates.**



By solving the above equation for *yf* ,



Using the above Equation 4.5.2, source *S* can calculate *yf* if it knows *xf* , which in turn can be calculated as follows. Using similarity of triangles again, the following equation is obtained:



By using the Pythagorean theorem,



Substituting for *yf* - *y*2 from [Equation i](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10equ03)n the above [Equation a](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10equ04)nd solving for *xf* , the following equation is obtained:



If updates include the direction information of nodes, only one previous update is required to predict future location (*xf , yf* ). The calculation of (*xf , yf* ) is then exactly the same as that of the periodic calculation of expected location (*xe , ye* ) by the update protocol.

**Delay Prediction**

The source node *S* has to predict the time instant *tf* at which a packet reaches the given destination node or intermediate node *D.* This can be known only if the end-to-end delay between nodes *S* and *D* is known. It is assumed that the end- to-end delay for a data packet from node *S* to node *D* is equal to the delay experienced by the latest update message received by node *S* from node *D.*

**QoS Routing**

Each node in the network has information about the complete network topology, which is refreshed by means of update messages. Using this information, the source node performs source-routing. The network state information is maintained in two tables, namely, the *update table* and the *routing table.* When node *A* receives an update message from node *B*, node *A* updates the corresponding entry for node *B* in the update table. In that entry, node *A* stores the ID of node *B*, the time instant at which the update packet was sent, the time at which the update packet was received, the geographic coordinates, speed, resource parameters of node *B*, and optionally the direction of motion of node *B.* For each node *N* in the network, node *A* stores the last two update packets received from that node in its update table. For some nodes, node *A* also maintains proximity lists. The proximity list of node *K* is a list of all nodes lying within a distance 1.5 × transmission range of node *K.* The proximity lists are used during route computation. By maintaining a proximity list rather than a neighbor list for node *K* (*i.e.,* list of nodes lying within node *K*'s transmission range), node *A* also considers the nodes that were outside node *K*'s transmission range at the time their respective last updates were sent, but that have since moved into node *K*'s transmission range, while computing the neighbors of node *K.* The routing table at node *A* contains information about all active connections with node *A* as source. When an update message from any node in the network reaches node *A*, it checks if any of the routes in its routing table is broken or is about to be broken. In either case, route recompilation is initiated. Using the location prediction based on the updates, it is possible to predict whether any link on the path is about to break. Thus, route recompilation can be initiated even before the route actually breaks.

The routing algorithm Works as follows. The source node *S* first runs location and delay predictions on each node in its proximity list in order to obtain a list of its neighbors at present. It determines which of these neighbors have the resources to satisfy the QoS requirements of the connection (the neighbors that satisfy the QoS requirements are called candidates). Then it performs a depth-first search for the destination, starting with each of these candidate neighbors to find all candidate routes satisfying the QoS requirements of the connection request. From the resulting candidate routes, the geographically shortest route is chosen and the connection is established. Data packets are forwarded along this chosen route until the end of the connection or until the route is recomputed in anticipation of breakage. Note that only node *S* uses its view of the network for the entire computation.

**Advantages and Disadvantages**

PLBQR protocol uses location and delay prediction schemes which reduce to some extent the problem arising due to the presence of stale routing information. Using the prediction schemes, it estimates when a QoS session will experience path breaks and proactively finds an alternate path to reroute the QoS session quickly. But, as no resources are reserved along the route from the source to the destination, it is not possible to provide hard QoS guarantees using this protocol. Even soft QoS guarantees may be broken in cases when the network load is high. Since the location prediction mechanism inherently depends on the delay prediction mechanism, the inaccuracy in delay prediction adds to the inaccuracy of the location prediction. The end-to-end delay for a packet depends on several factors such as the size of the packet, current traffic load in the network, scheduling policy and processing capability of intermed iate nodes, and capacity of links. As the delay prediction mechanism does not take into consideration some of the above factors, the predictions made by the location prediction mechanism may not be accurate, resulting in QoS violatio ns for the real-time traffic.

***4.5.4 Trigger-Based Distributed QoS Routing Protocol***

The trigger-based (on-demand) distributed QoS routing (TDR) protocol was proposed by De *et al.*for supporting real-time applications in ad hoc wireless networks. It operates in a distributed fashion. Every node maintains only the local neighborhood informatio n in order to reduce computation overhead and storage overhead. To reduce control overhead, nodes maintain only the active routes. When a link failure is imminent, TDR utilizes the global positioning system-based (GPS) location informatio n of the destination to localize the reroute queries only to certain neighbors of the nodes along the source-to-destination active route. For a quick rerouting with reduced control overhead, rerouting is attempted from the location of an imminent link failure, called intermed iate node-initiated rerouting (INIR). If INIR fails, then in order to keep the flow state disruption to a minimum, rerouting is attempted from the source, which is termed source-initiated rerouting (SIRR).

**Database Management**

All nodes in the network maintain the local neighborhood information. For each neighbor, every node maintains *received power level,* current geographic coordinates, velocity, and direction of motion in the database.

**Activity-Based Database**

In addition to the local neighborhood information, node *N* maintains a source table *ST N* , a destination table *DT N* , or an intermediate table *IT N* based on whether it actively participates in a session as the source (*S*), the destination (*D*), or as an intermediate node (*I*), respectively. These tables are referred to as the activity- based database. For a session, the source table contains the following fields: session ID, source ID, destination ID, maximum bandwidth demand (MaxBW), maximum acceptable delay (MaxDelay measured in terms of hop count), destination location (DLoc), next-node ID (NID) toward the destination, and activity flag (NodActv). An intermediate table contains the following fields: session ID, source ID, destination ID, source location (SLoc), MaxBW, MaxDelay, DLoc, NID, previous-node ID toward the source (PID), distance from the source (measured in terms of hop count), and NodActv. The destination table contains the following fields: session ID, source ID, destination ID, SLoc, MaxBW, MaxDelay, PID, distance from the source (hop count), and NodActv. At any time instant, a node may have to maintain one or more tables simultaneo usly for different on-going sessions. Each node *N* also maintains an updated residual bandwidth (*ResiBWN* ) which indicates its ability to participate in a session. A soft state approach is used to maintain the activity-based database. Hence, the database needs to be refreshed periodically. It is refreshed when data packets belonging to the on-going sessions are received by a node.

**Routing Protocol**

The messages that are exchanged for initiating, maintaining, and terminating a real-time session are described below. Initial Route Discovery If the source *S* has enough *Resi BWS* to satisfy the Max BW for the session, the required bandwidth is temporarily reserved for a certain duration within which it expects an acknowledgment from the destination *D.* If the source knows the location of the destination, it performs route discovery through selective forwarding. In this approach, the source node takes advantage of location information of its neighbors and forwards route requests to only selective neighbors that are lying closely toward the destination node and satisfying QoS requirements of the connection request. Otherwise, the source initiates a flooding-based initial route discovery process. Before transmitting the route discovery packet, an entry is made in the source table *ST S* for this session with NodActv flag set to zero (*i.e*., idle). To ensure the stability of routes and in order to reduce the control overhead, only selected neighbors, from which packets were received with power level more than a threshold level (*P th* 1 ), are considered during route establishment. After receiving a route discovery packet, an intermediate node (IN) checks in its *IT IN* whether any such packet was already received for the same session. If so, the current route discovery packet is rejected to ensure loop-free routing. Otherwise, it is the first discovery packet for a session. Then the intermediate node (IN) increments the hop-count field of the received packet by one and checks for *ResiBWIN* . If it can meet the MaxBW requirement for the session and if the updated hop-count field is less than MaxDelay, the required bandwidth is temporarily reserved, and an entry is made into the activity table *IT IN* for the session with NodActv flag set to zero. Then the packet is forwarded to its downstream neighbors with the updated NID field. If either or both of *ResiBW* and MaxDelay criteria cannot be satisfied, the discovery packet is simply dropped. Upon receiving the first discovery packet, if the destination *D* is also able to satisfy both the *ResiBW* and the MaxDelay criteria, the discovery packet and the corresponding route are accepted.

**Route/Reroute Acknowledgment**

After accepting the route, the destination node *D* builds *DT D* table with the NodActv flag set to 1 (*i.e.,*active) and sends an ACK to the source *S* along the selected route. On receiving the ACK packet, all intermediate nodes and the source *S* set the NodActv flags in their respective tables to 1 and refresh their *ResiBW* status. The packet transmission for the session follows immediately.

**Alternate Route Discovery**

In SIRR, when the received power level at an intermediate node falls below a threshold *P th* 2 , the intermediate node sends a rerouting indication to the source *S*. Then the source *S* initiates the rerouting process through selective forwarding. But in INIR, when the power level of a packet received from the next node toward the destination falls below a threshold *P th* 1 (*P th* 1*> P th* 2 ), it initiates a status query packet toward the source with appropriate identification fields and with a flag field called route repair status (*RR\_Stat*) set to zero. If any upstream node is in the rerouting process, upon reception of the status query packet it sets the *RR\_Stat* flag to 1 and sends the status reply packet to the querying node. On arriving at the source, the status query packet is discarded. If the querying node receives no status reply packet before its received power level from the downstream node goes below *P th* 2 , it triggers the alternate route discovery process (*i.e.,* SIRR). Otherwise, it relinquishes control of rerouting. This query/reply process eliminates the chances of duplicate reroute discovery for a session. In both SIRR and INIR, the alternate route discovery process is similar to the initial route discovery except that the rerouting process takes advantage of the location information of the local neighbors and the approximate location of the destination, and forwards the rerouting requests to only selected neighbors that are close to the destination and that satisfy the delay and bandwidth constraints. The threshold parameters *P th* 1 and *P th* 2 have to be selected judiciously in order to avoid unnecessary rerouting.

**Route Deactivation**

In case of session completion or termination, the source node purges its corresponding *ST* table and sends a route deactivation packet toward the destination. Upon receiving a deactivation request, each node which was part of that session updates its *ResiBW* and purges the activity table for that session. No explicit deactivation packet is sent in case of rerouting, as the new route could still consist of some nodes that were part of the old route.

**Advantages and Disadvantages**

In TDR protocol, if the source node knows the location of the destination node, it performs route discovery through selective forwarding to reduce the control overhead. For a quick rerouting with reduced control overhead and to reduce the packet loss during path breaks, it uses INRR and SIRR schemes. However, in this protocol a QoS session is rerouted if the received power level from a downstream node falls below a certain value (*i.e.,* threshold). Due to small-scale fading, the received power level may vary rapidly over short periods of time or distance traveled. Some of the factors that influence fading are multipath propagation, velocity of the nodes, and bandwidth of the channel. Even though the downstream node may be within the transmission range of the upstream node, due to fading the received power level at the upstream node may fall below the threshold value. This increases the control overhead because of initiation of the alternate route discovery process and false rerouting of some of the sessions.

***4.5.5 QoS-Enabled Ad Hoc On-Dem and Distance Vector Routing Protocol***

Perkins *et al.* have extended the basic ad hoc on-demand distance vector (AODV) routing protocol to provide QoS support in ad hoc wireless networks. To provide QoS, packet formats have been modified in order to specify the service requirements which must be met by the nodes forwarding a *RouteRequest* or a *RouteReply.*

**QoS Extensions to AODV Protocol**

Several modificatio ns have been carried out for the routing table structure and *RouteRequest* and*RouteReply* messages in order to support QoS routing. Each routing table entry corresponds to a different destination node. The following fields are appended to each routing table entry:

• Maximum delay

• Minimum availab le bandwidth

• List of sources requesting delay guarantees

• List of sources requesting bandwidth guarantees

**Maximum Delay Extension Field**

The maximum delay extension field is interpreted differently for *RouteRequest* and *RouteReply*messages. In a *RouteRequest* message, it indicates the maximum time (in seconds) allowed for a transmission from the current node to the destination node. In a *RouteReply* message, it indicates the current estimate of cumulative delay from the current intermed iate node forwarding the *RouteReply,*to the destination. Using this field the source node finds a path (if it exists) to the destination node satisfying the maximum delay constraint. Before forwarding the *RouteRequest,* an intermed iate node compares its *node traversal time* (*i.e.,* the time it takes for a node to process a packet) with the (remaining) delay indicated in the maximum delay extension field. If the delay is less than the node traversal time, the node discards the *RouteRequest* packet. Otherwise, the node subtracts node traversal time from the delay value in the extension and processes the *RouteRequest* as specified in theAODV protocol. The destination node returns a *RouteReply* with the maximum delay extension field set to zero. Each intermed iate node forwarding the *RouteReply* adds its own node traversal time to the delay field and forwards the *RouteReply* toward the source. Before forwarding the *RouteReply* packet, the intermed iate node records this delay value in the routing table entry for the corresponding

destination node.

**Minimum Bandwidth Extension Field**

In a *RouteRequest* message, this field indicates the minimum bandwidth (in Kbps) that must be availab le along an acceptable path from the source to the destination. In a *RouteReply* message, it indicates the minimum bandwidth availab le on the route between the node forwarding the *RouteReply*and the destination node. Using this field, the source node finds a path (if it exists) to the destination node satisfying the minimum bandwidth constraint. Before forwarding the *RouteRequest,* an intermed iate node compares its availab le bandwidth with the bandwidth field in the extension. If the requested amount of bandwidth is not availab le, the node discards the *RouteRequest* message. Otherwise, the node processes the *RouteRequest* as specified in the AODV protocol. The destination node returns a *RouteReply* in response to a *RouteRequest* with the bandwidth field set to infinity (a very large number). Each node forwarding the *RouteReply* compares the bandwidth field in the *RouteReply* with its own link capacity and updates the bandwidth field of the *RouteReply* with the minimum of the two, before forwarding the *RouteReply.* This value is also stored in the routing table entry for the corresponding destination and indicates the minimum availab le bandwidth to the destination.

List of Sources Requesting QoS Guarantees.A *QoSLost* message is generated when an intermed iate node experiences an increase in node traversal time or a decrease in the link capacity. The *QoSLost* message is forwarded to all sources potentially affected by the change in the QoS parameter. These are the sources to which *RouteReply*s with QoS extension have been forwarded by the node earlier.

**Advantages and Disadvantages**

The advantage of QoS AODV protocol is the simplicity of extension of the AODV protocol that can potentially enable QoS provisioning. However, as no resources are reserved along the path from the source to the destination, this protocol is not suitable for applications that require hard QoS guarantees. Further, node traversal time is only the processing time for the packet, so the major part of the delay at a node is contributed by packet queuing and contention at the MAC layer. Hence, a packet may experience much more delay than this when the traffic load is high in the network.

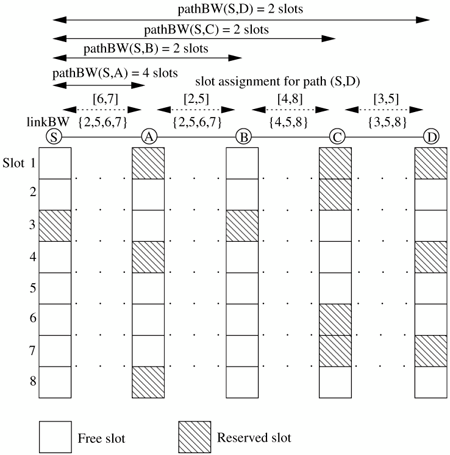
**4.5.6 Bandwidth Routing Protocol**

The bandwidth routing (BR) protocol consists of an end-to-end path bandwidth calculation algorithm to inform the source node of the available bandwidth to any destination in the ad hoc network, a bandwidth reservation algorithm to reserve a sufficient number of free slots for the QoS flow, and a standby routing algorithm to reestablish the QoS flow in case of path breaks. Here, only bandwidth is considered to be the QoS parameter. In TDMA-based networks, bandwidth is measured in terms of the number of free slots available at a node. The goal of the bandwidth routing algorithm is to find a shortest path satisfying the bandwidth requirement. The transmission time scale is organized into frames, each containing a fixed number of time-slots. The entire network is synchronized on a frame and slot basis. Each frame is divided into two phases, namely, the control phase and the data phase. The control phase is used to perform the control functions such as slot and frame synchronization, virtual circuit (VC) setup, and routing. The data phase is used for transmission/reception of data packets. For each node, a slot is assigned in the control phase for it to broadcast its routing information and slot requirements. At the end of the control phase, each node knows about the channel reservations made by its neighbors. This information helps nodes to schedule free slots, verify the failure of reserved slots, and drop expired real-time packets. The BR protocol assumes assumes a half-duplex CDMA-over-T DMA system in which only one packet can be transmitted in a given slot. Bandwidth Calculation Since the network is multi-hop in nature, the free slots recorded at each node may be different. The set of common free slots between two adjacent nodes denotes the link bandwidth between them. The path bandwidth between two nodes is the maximum bandwidth available in the path between them. If the two nodes are adjacent, the path bandwidth between them equals their link bandwidth. For example, consider two adjacent nodes, node A and node B, having free slots {2,5,6,8} and {1,2,4,5}, respectively. The link bandwidth *linkBW* (*A, B*) = *freeslot*(*A*) ∩ *freeslot* (*B*) = {2, 5}. It means that only slots 2 and 5 can be used by nodes A and B for transmitting data packets to each other. The *freeslot* (*X*) is defined as the set of slots which are not used by any adjacent node of node *X* (to receive or to send) from the point of view of node *X*. To compute the end-to-end bandwidth for a path in a TDMA-based network, one has to know not only the available bandwidth on the individ ual links on the path, but also determine the scheduling of the free slots. The BR protocol also provides a heuristic-based hop-by-hop path bandwidth calculation algorithm to assign free slots at every hop along the path. The call admission control mechanism of theBR protocol uses the information regarding the availab ility of end-to-end bandwidth while making a decision on whether to admit or reject a new QoS session. The path bandwidth calculation algorithm is explained with the help of the example shown in [Figure 4.10, w](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10fig10)here a path from source node *S* to destination node *D* is illustrated. The process of computing *pathBW*(*S, D*) is explained below.

• *pathBW*(*S, A*): Since node *S* and node *A* are adjacent, the *pathBW*(*S, A*) = *linkBW*(*A, S*), which is four slots. The four slots are {2, 5, 6, 7}.

• *pathBW*(*S, B*): Since *pathBW*(*S, A*) = *linkBW*(*A, B*) = {2, 5, 6, 7}, if *S* uses slots 6 and 7 to send packets to *A*, then *A* can use only slots 2 and 5 for transmission of packets to *B.* This is because a node cannot be in transmission and reception modes simultaneous ly. Hence *pathBW*(*S, B*) is two slots, by assigning slots {6, 7} on link(*S*, *A*) and slots {2, 5} on link(*A*, *B*).

**Figure 4.10. An example of path bandwidth calculation in BR protocol.**



• *pathBW*(*S, C*): Here slots 4 and 8 are exclusively availab le for *linkBW*(*B, C*), slot 2 is exclusively availab le for *pathBW*(*S, B*), and slot 5 is common for both of them. So assign one of slots 4, 8 to link(*B, C*), for example, assign slot 4 to link(*B, C*), and slot 2 to path(*S, B*). For achieving maximum bandwidth, assign slot 8 to link (*B, C*) and slot 5 to path(*S, B*). Hence, *pathBW*(*S, C*) is 2 slots, by assigning slots {6, 7} on link(*S, A*), slots {2, 5} on link(*A, B*), and slots {4, 8} on link(*B, C*).

• *pathBW*(*S, D*): This case is similar to the previous one. So slots 4 and 8 are assigned to path(*S, C*) and slots 3 and 5 are assigned to link(*C, D*) to get two slots for *pathBW*(*S, D*).

**Slot Assignment**

The path bandwidth calculation algorithm requires periodic exchange of bandwidth information. The slot assignment algorithm in each node assigns free slots during the call setup. When a node receives a call setup packet, it checks whether the slots that the immediate sender will use for transmission are free, and it also finds out if there are free slots that can be used for forwarding the incoming packets. If such free slots are available, the slot assignment algorithm reserves the required number of slots, updates the routing table, and then forwards the call setup packet to the next hop. If the required number of slots are not available at the node, all the reservations that have been made so far along the path from the source node to the current node have to be canceled in order to release the slots assigned for this connection. This is done by sending a *Reset* packet back to the source along the path that has been established so far. If reservations are made successfully along the path from the source to the destination, the destination sends a *Reply* packet back to the source to acknowledge having set up the connection. The reservations are soft state in nature in order to avoid resource lock-up at intermediate nodes due to path breaks.

**Standby Routing Mechanism**

The connections may get broken due to dynamic changes in the network topology. The standby routing mechanism has to reestablish such broken connections. Secondary paths are maintained in the routing table, which can be used when the primary path fails. The standby route is easily computed using the DSDV algorithm [w](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10biblio01entry19)ithout any extra overhead. Each node periodically exchanges routing information with its neighboring nodes. The neighbor with the shortest distance to the destination node becomes the next node on the primary path to the destination node. The neighbor node with the second shortest distance to the destination becomes the next node on the standby route to the destination. It is to be noted that this standby route is not guaranteed to be a link- or node-disjoint one. When a primary path fails, the upstream node that detects the link break will try to rebuild a new path immediately, using the standby route. If the standby route satisfies the QoS requirements, the new path from the point of the path break is established by sending a call setup packet hop-by-hop to the destination through the standby path. Since this scheme follows DSDV protocol, a table-driven routing protocol, and uses on-demand call admission control, similar to the on-demand routing

protocols, it is classified into the category of hybrid solutions in the classifications in Figure

**Advantages and Disadvantages**

The BR protocol provides an efficient bandwidth allocation scheme for CDMA- over-TDMA-bas ed ad hoc wireless networks. The standby routing mechanism can reduce the packet loss during path breaks. But the CDMA-over- TDMA channel model that is used in this protocol requires assigning a unique control slot in the control phase of super frame for each node present in the network. This assignment has to be done statically before commissioning the network. Due to this, it is not possible for a new node to enter into the network at a later point of time. If a particular node leaves the network, the corresponding control slot remains unused and there is no way to reuse such a slot(s). Further, the network needs to be fully synchronized.

***4.5.7 On-Dem and QoS Routing Protocol***

Lin proposed an admission control scheme over an on-demand QoS routing (OQR) protocol to guarantee bandwidth for real-time applications. Since routing is on-demand in nature, there is no need to exchange control information periodically and maintain routing tables at each node. Similar to the bandwidth routing (BR) protocol, the network is time-slotted and bandwidth is the key QoS parameter. The path bandwidth calculation algorithm proposed in BR is used to measure the availab le end-to-end bandwidth. The on-demand QoS routing protocol is explained below.

**Route Discovery**

During the route discovery process, the source node that wants to find a QoS route to the destination floods a QoS route request (QRREQ) packet. A QRREQ packet contains the following fields: packet type, source ID, destination ID, sequence number, route list, slot array list, data, and TTL. The pair {source ID, sequence number} is used to uniquely identify a packet. For each QRREQ packet, the source node uses a new sequence number (which is monotonically increasing) in order to avoid multiple forwarding of the same packet by intermed iate nodes. The route list records the nodes that have been visited by the QRREQ packet, whereas the slot array list records free slots availab le at each of these nodes. The TTL field limits the maximum length of the path to be found. A node *N* receiving aQRREQ packet performs the following operations:

1. If a QRREQ with the same {source ID, sequence number} had been received already, this QRREQ packet gets discarded.

2. Otherwise, the route list field is checked for the address of this node *N.* If it is present, node *N*discards this QRREQ packet.

3. Otherwise,

• Node *N* decrements TTL by one. If TTL counts down to zero, it discards this QRREQ packet.

• It calculates the path bandwidth from the source to this node. If it satisfies the QoS requirement, node *N* records the available free slots in the slot array list of the QRREQ packet. Otherwise, node *N* discards this QRREQ packet.

• Node *N* appends the address of this node to the route list and re- broadcasts this QRREQ packet if it is not the destination.

For the example shown in [Figure 4.10, a](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10fig10)ssume that the source *S* floods a QRREQ packet with bandwidth requirement of two time-slots. Here, the destination *D* receives a QRREQ packet with the following information in its fields. The route list field contains (*S, A, B, C*) and the slot array list contains ([A, {2, 5, 6, 7}], [B, {2, 5}], [C, {4, 5}], [D, {3, 8}]). The destination may receive more than one QRREQ packet, each giving a unique feasible QoS path from the source to the destination.

**Bandwidth Reservation**

The destination node may receive one or more QRREQ packets, each giving a feasible QoS path for the connection request. The destination node selects the least-cost path among them. Then it copies the fields {route list, slot array list} from the corresponding QRREQ packet to the QoS Route Reply (QRREP) packet and sends the QRREP packet to the source along the path recorded in the route list. As the QRREP traverses back to the source, each node recorded in the route list reserves the free slots that have been recorded in the slot array list field. Finally, when the source receives the QRREP, the end-to-end bandwidth reservation process is completed successfully. The reservations made are soft state in nature in order to avoid resource lock-up. The source can start sending data packets in the data phase. At the end of the session, all reserved slots are released.

**Reservation Failure**

The reservation of bandwidth may fail, either due to route breaks or because the free slots that are recorded in the slot array list get occupied by some other connection(s) before the QRREP packet sent by the destination reaches the corresponding intermed iate nodes. In the second case, the node at which reservation fails, sends a *Reserve Fail* packet to the destination node. The destination then restarts the reservation process along the next feasible path. All nodes on the path from the interrupted node to the destination free the reserved slots for this connection on receiving the *Reserve Fail* packet. If no connection could be set up due to non availab iliy of feasible paths, the destination broadcasts a *NoRoute* packet to notify the source. Then the source either restarts the route discovery process, if it still needs a connection to the destination, or rejects the call.

**Route Maintenance**

When a route gets broken, the nodes detecting the link break send a *RouteBroken* packet to the source and the destination nodes. In other words, once the next hop becomes unreachable, the upstream node which is toward the source node sends a *RouteBroken* packet to the source, and the downstream node which is toward the destination sends another *RouteBroken* packet to the destination. The intermed iate nodes relaying the *RouteBroken* packet release all reserved slots for this connection and drop all data packets of this connection which are still pending in their respective queues. After receiving the *RouteBroken* packet, the source restarts the route discovery process in order to reestablish the connection over a new path, while the destination releases resources reserved for that connection.

**Advantages and Disadvantages**

OQR protocol uses an on-demand resource reservation scheme and hence produces lower control overhead. Since it uses the CDMA-over-T DMA channel model, the network needs to be fully synchronized. Further, the on-demand nature of route discovery process leads to higher connection setup time.

***4.5.8 On-Dem and Link-State Multipath QoS Routing Protocol***

Unlike the QoS routing protocols described above in this chapter which try to find a single path from the source to the destination satisfying the QoS requirements, the on-demand link-state multipath QoS routing (OLMQR) protocol searches for multiple paths which collectively satisfy the required QoS. The original bandwidth requirement is split into sub-bandwidth requirements. Notably, the paths found by the multipath routing protocol are allowed to share the same sub-paths. OLMQR has better call acceptance rate in ad hoc wireless networks where finding a single path satisfying all the QoS requirements is very difficult. In this protocol, the MAC layer is assumed to be using the CDMA-over- TDMA channel model similar toBR and OQR protocols. A mobile node in the network knows the bandwidth availab le to each of its neighbors. When the source node requires a QoS session with bandwidth *BW* to the destination, it floods a QoS route request (QRREQ) packet. Each packet carries the path history and link-state information from the source to the destination. The destination node collects all possible link-state information from different QRREQ packets received and constructs its own view of the current network topology. A multipath routing algorithm is applied at the destination to determine multip le paths which collectively fulfill the original bandwidth requirement *BW* of the QoS flow. Then the destination node sends reply packets along these paths, which reserve the corresponding resources (sub-bandwidth requirements) on the corresponding paths on their way back to the source. The operation of this protocol consists of three phases: Phase 1 is on-demand link-state discovery, phase 2 is unipath discovery, and phase 3 is multipath discovery and reply.

**On-Demand Link-State Discovery**

For each call request, the source node floods a QRREQ packet toward the destination. Each packet records the path history and all link-state information along its route. A QRREQ packet contains the following fields: source ID, destination ID, node history, free time-slot list, bandwidth requirement, and time to live (TTL). The node history field records the path from source to the current traversed node, the free time-slot list field contains a list of free time-slots of links, where each entry in the list records free time-slots between the current traversed node and the last node recorded in the node history, and TTL field limits the hop length of the search path. The source *S* floods a QRREQ(*S*, *D,* node history = {*S*}, free time-slot list = φ, BW, TTL) packet into the network toward the destination D, if the given requirement is *BW.* An intermed iate node *N* receiving aQRREQ packet performs the following operations:

1. Node *N* checks the node history field of the QRREQ packet for its address. If it is present, the node discards this QRREQ packet.

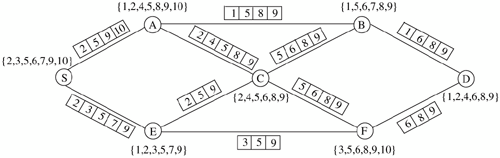
2. Otherwise,

• Node *N* decrements TTL by one. If TTL counts down to zero, it discards this QRREQ packet.

• Node *N* adds itself into the node history field, appends the free time- slots of the link between itself and the last node recorded in the node history field into the free time-slot list field, and rebroadcasts this QRREQ packet. The destination may receive many different QRREQ packets from the source. It constructs its own view of the current network topology. It also calculates the available bandwidths of the links present in that network topology. For example, consider the network shown in [Figure 4.11. T](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10fig11)he source *S* floods the network with a QRREQ packet by setting *BW* and *TTL* fields to 3 and 4, respectively.

The destination *D*receives six QRREQ packets, which have traversed along the paths: *S* → *A* → *B* → *D, S* → *E* → *F* → *D, S → A → C → B → D, S* → *A* → *C* → *F* → *D, S → E → C → F → D, and S → E → C → B →D.* Using this information, a partial view of the network is constructed at the destination *D*.

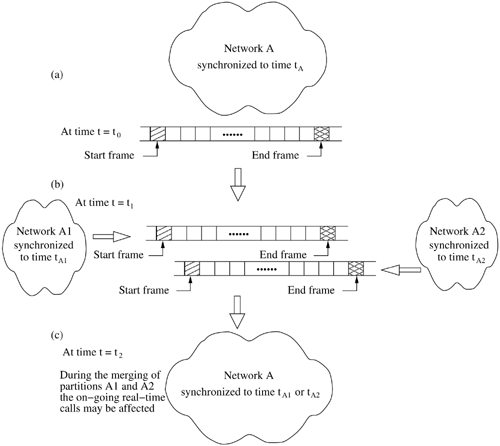
**Figure 4.11. An example network.**



***4.5.9 Asynchronous Slot Allocation Strategies***

The QoS solutions discussed so far such as BR, OQR, and OLMQR assume a TDMA-based network or aCDMA-over-T DMA model for the network. This requires time synchronization across all nodes in the network. Time synchronization demands periodic exchange of control packets, which results in high bandwidth consumption. Ad hoc wireless networks experience rapid changes in topology leading to a situation where network partitions and merging of partitions can take place. [Figure 4.12 s](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10fig14)hows the synchronization problems arising out of dynamic topological changes in an ad hoc wireless network.

**Figure 4.12. Illustration of synchronization problems in a dynamic network topology.**



A completely connected and synchronized network A at time *t* = *t*0 (shown in Figure 4.12 (a)) may be partitioned into two disjoint networks A1 and A2 at time *t* = *t*1 (shown in Figure 4.12 (b)). These two networks may be synchronized to two different clock times as illustrated. Due to the dynamic topology experienced in an ad hoc wireless network, it is possible to have two separately synchronized networks A1 (synchronized to *tA* 1 ) and A2 (synchronized to *tA* 2 ) merge to form a combined network A ([Figure 4.12 (c)).](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10fig14) During the merging process, the real-time calls existing in the network may be affected while accommodating the changes in synchronization. The asynchronous QoS routing (AQR) scheme and slot allocation strategies proposed provide a unique mechanism to reserve asynchronous end-to-end bandwidth for real-time calls in ad hoc wireless networks. The three major phases in the operation of AQR are bandwidth feasibility test phase, bandwidth allocation phase, and bandwidth reservation phase. An in-depth discussion of each of these phases follows. Bandwidth Feasibility Test Phase The objective of this phase is the selection of paths with required bandwidth, which is achieved by the propagation of *RouteRequest* packets. The source node that needs to set up a QoS path to a destination originates *RouteRequest* packets addressed to the destination. An intermed iate node that receives this*RouteRequest* checks for bandwidth availab ility in the link through which it received the *RouteRequest*packet. AQR interacts with the MAC layer for obtaining reservation information. If sufficient bandwidth is availab le, then it forwards the *RouteRequest* packet, else the packet is dropped. The intermed iate node adds its own reservation table along with the reservation tables of the nodes the packet has already traversed before forwarding it further. Routing loops are avoided by keeping track of the sequence number, source address, and traversed path informations contained in the *RouteRequest*packet. Apart from this reservation table, an intermed iate node also incorporates necessary information in an *offset time* field to enable the destination node to make use of the reservation table. In other words, the offset time field carries synchronization information required for interpreting the reservation table with respect to the receiving node's current time. When the source node constructs a*RouteRequest* packet, it stores its reservation table in the packet with respect to its current time with the quantity offset set to zero. When the packet is about to be sent, the difference between the current time and time of construction of packet is stored in the offset. When the *RouteRequest* packet is received at a node, the offset is increased by the estimated propagation delay of transmission. Hence by using this offset time, the relative difference between the local clock and the time informatio n contained in the reservation table carried in the *RouteRequest* can be incorporated and then used for synchronizing the reservation information. When the *RouteRequest* packet reaches its destination, it runs the slot allocation algorithm on a selected path, after constructing a data structure called *QoS Frame* for every link in that path. The *QoS Frame* is used to calculate, for every link, the free bandwidth slots in the super frame and unreservable slots due to reservations carried out by the neighborhood nodes (also referred to as unreservab le slots due to hidden terminals). The destination node waits for a specific time interval, gathers a set of *RouteRequest* packets, and chooses a shortest path with necessary bandwidth.

**Bandwidth Allocation Phase**

In this phase, the destination node performs a bandwidth allocation strategy that assigns free slots to every intermed iate link in the chosen path. The information about asynchronous slots assigned at every intermed iate link is included in the *RouteReply* packet and propagated through the selected path back to the source. Slot allocation strategies such as early fit reservation (EFR), minimum bandwidth-based reservation (MBR), position-based hybrid reservation (PHR), and *k*-hopcount hybrid reservation (*k*-HHR) discussed later in this section can be used for allocation of bandwidth and positioning of slots in this phase.

**Slot Allocation Strategies**

The slot allocation strategies are used in the bandwidth allocation phase in order to decide upon the order of links in a chosen path and particular slot positions to be assigned. The order of links chosen for allocation and the position of assigned bandwidth-slots on each link influence the end-to-end delay of the path and the call acceptance rate. • Early fit reservation (EFR): During the bandwidth allocation phase, the destination node runs the following steps for the EFR scheme:

Step 1: Order the links in the path from source to destination.

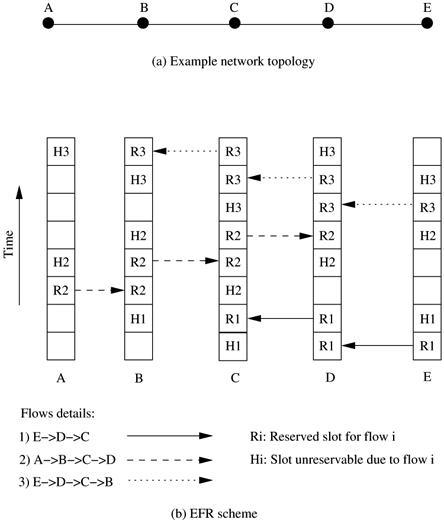
Step 2: Allocate the first available free slot for the first link in the path.

Step 3: For every subsequent link, allocate the first immediate free slot after the assigned slot in the previous link.Step 4: Continue Step 3 until the last link in the chosen path is reached. EFR attempts to provide the least end-to-end delay. The average end-to-end delay

can be obtained as 

where *n* is the number of hops in the path and *tsf* is the duration of the super frame. Figure 4.13 (a) illustrates a simple string topology and Figure 4.13 (b) shows the slot allocation carried out for three real-time flows. In the example, the average delay experienced can be calculated as  slots. The flow *E* → *C* experiences a delay of two slots, and flows *A* → *D* and *E* → *B* experience a delay of three slots each, making the average delay of  slots.

**Figure 4.13. Illustration of EFR scheme.**



• Minimum bandwidth-based reservation (MBR): The following steps are executed by the destination node for the MBR scheme:

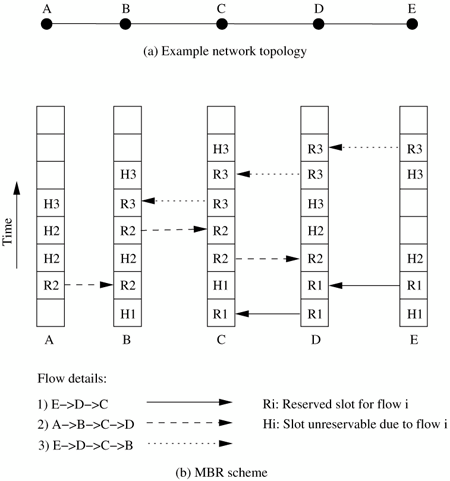
Step 1: Order the links in the non-decreasing order of free bandwidth.

Step 2: Allocate the first free slot in the link with lowest free bandwidth.

Step 3: Reorder the links in the non-decreasing order of free bandwidth and assign the first free slot on the link with lowest bandwidth.

Step 4: Continue Step 3 until bandwidth is allotted for all the links.

MBR allots bandwidth for the links in the increasing order of free bandwidth. In case a tie occurs, where two links exist with the same amount of free bandwidth, it is broken by choosing the link with lowest bandwidth in the neighboring links. Further ties are broken by choosing the link with lowest ID of the link-level sender. Figure 4.14 (b) shows the slot allocation carried out in the MBR scheme over a simple string topology network. The worst case end-to- end delay provided by MBR can be (*n* - 1) × *tsf* . In the example in Figure 4.14 [(b),](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch10.html%23ch10fig16) the average delay experienced can be calculated as  slots.

**Figure 4.15. Illustration of MBR scheme.**

• Position-based hybrid reservation (PHR): Similar

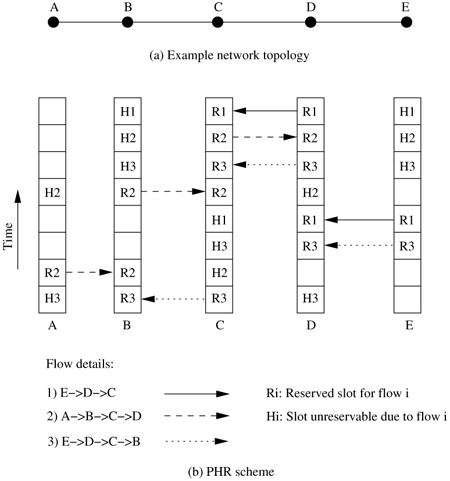
to EFR and MBR schemes, PHR also is executed at the destination node. The following are the steps in the PHR algorithm :

Step 1: List the links in the order of increasing bandwidth.

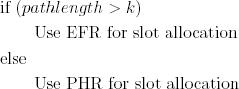
Step 2: Assign a free slot for the link with least amount of bandwidth, such that the position of assignment of bandwidth is proportional to  where *i* is the position of the link and *L path* is the path length.

Step 3: Repeat Step 2 until all the links are assigned with free slots. Figure 4.15 shows the slot allocation done on a string topology for three flows. In the given example, the average delay experienced can be calculated as  slots.

**Figure 4.16. Illustration of PHR scheme.**



• *k*-hopcount hybrid routing (*k*-HHR): This is a hybrid slot allocation scheme in which either EFR orPHR is chosen dynamically by the destination node based on the hop length of the path. The *k*-HHR scheme is described below.



This takes the end-to-end delay advantage of the EFR scheme for long flows

and the high call acceptance with medium end-to-end delay of the PHR scheme for flows with shorter length.

**Bandwidth Reservation Phase**

In this phase, a reservation of bandwidth at every link of a path is carried out. The reservation is effected by the intermediate nodes with the information carried in the *RouteReply* packet, in an asynchronous fashion using RTMAC protocol. Once the reservation at an intermediate link is successful in the designated time duration (the time duration for a free conn- slot, at which the reservation is to be carried out), the *RouteReply* packet is further forwarded. If the designated slot is not free at the time the intermed iate node attempts the reservation (this can happen either due to the mobility of nodes or due to the staleness of the information), the intermed iate node can try reserving any of the free slots availab le. If the intermed iate node finds it impossible to reserve bandwidth, it drops the *RouteReply* and sends a control packet to the destination, which makes all the nodes in its way, those that have successfully reserved bandwidth, release the bandwidth and the destination node to find another path with the necessary bandwidth.

**Advantages and Disadvantages**

AQR has a unique advantage in that it can provide end-to-end bandwidth reservation in asynchronous networks. Also, the slot allocation strategies can be used to plan for the delay requirements and dynamically choose appropriate algorithms. AQR is an on-demand QoS routing scheme and hence the setup time and reconfiguration time of real-time calls are high. Also, the bandwidth efficiency of such an asynchronous system may not be as high as a fully synchronized TDMA system due to the formation of bandwidth holes (short free slots which cannot be used).

**4.6 SUMMARY**

In this chapter, several solutions proposed in the literature for providing QoS support for applications in ad hoc wireless networks have been described. First, the issues and challenges in providing QoS in ad hoc wireless networks were discussed. Then the classifications of the existing QoS approaches under several criteria such as interactio n between routing protocol and resource reservation signaling, interaction between network and MAC layer, and informatio n update mechanism were discussed. The data link layer solutions such as cluster TDMA, IEEE 802.11e, and DBASE and the network layer solutions such as ticket-based probing, predictive location-based QoS routing, trigger- based QoS routing, QoS enabled AODV, bandwidth routing, on-demand routing, asynchronous QoS routing, and multipath QoS routing were described. Finally, QoS frameworks for ad hoc wireless networks such as INSIGNIA, INORA, SWAN, and PRTMAC were described.

**4.2 ENERGY MANAGEMENT IN AD HOC WIRELESS NETWORKS**

**4.2.1 INTRODUCTION**

The nodes in an ad hoc wireless network are constrained by limited battery power for their operation. Hence, energy management is an important issue in such networks. The use of multi-hop radio relaying requires a sufficient number of relaying nodes to maintain the network connectivity. Hence, battery power is a precious resource that must be used efficiently in order to avoid early termination of any node.

Energy management deals with the process of managing energy resources by means of controlling the battery discharge, adjusting the transmission power, and scheduling of power sources so as to increase the lifetime of the nodes of an ad hoc wireless network. Efficient battery management, transmission power management, and system power management are the three major means of increasing the life of a node. Battery management is concerned with problems that lie in the selection of battery technologies, finding the optimal capacity of the battery, and scheduling of batteries, that increase the battery capacity. Transmission power management techniques attempt to find an optimum power level for the nodes in the ad hoc wireless network. On the other hand, system power management deals mainly with minimizing the power required by hardware peripherals of a node (such as CPU, DRAM, and LCD display) and incorporating low-power strategies into the protocols used in various layers of the protocol stack. This chapter concentrates on the issues involved and the solutions for energy management in ad hoc wireless networks.

**4.2.2 NEED FOR ENERGY MANAGEMENT IN AD HOC WIRELESS NETWORKS**

The energy efficiency of a node is defined as the ratio of the amount of data delivered by the node to the total energy expended. Higher energy efficiency implies that a greater number of packets can be transmitted by the node with a given amount of energy reserve. The main reasons for energy management in ad hoc wireless networks are listed below:

• **Limited energy reserve:** The main reason for the development of ad hoc wireless networks is to provide a communication infrastructure in environments where the setting up of a fixed infrastructure is impossible. Ad hoc wireless networks have very limited energy resources. Advances in battery technologies have been negligible as compared to the recent advances that have taken place in the field of mobile computing and communication. The increasing gap between the power consumption requirements and power availability adds to the importance of energy management.

• **Difficulties in replacing the batteries:** Sometimes it becomes very difficult to replace or recharge the batteries. In situations such as battlefields, this is almost impossible. Hence, energy conservation is essential in such scenarios.

• **Lack of central coordination:** The lack of a central coordinator, such as the base station in cellular networks, introduces multi-hop routing and necessitates

that some of the intermediate nodes act as relay nodes. If the proportion of relay traffic is large, then it may lead to a faster depletion of the power source for that node. On the other hand, if no relay traffic is allowed through a node, it may

lead to partitioning of the network. Hence, unlike other networks, relay traffic plays an important role in ad hoc wireless networks.

• **Constraints on the battery source:** Batteries tend to increase the size and weight of a mobile node. Reducing the size of the battery results in less capacity which, in turn, decreases the active lifespan of the node. Hence, in addition to reducing the size of the battery, energy management techniques are necessary to utilize the battery capacity in the best possible way.

• **Selection of optimal transmission power:** The transmission power selected determines the reach ability of the nodes. The consumption of battery charge increases with an increase in the transmission power. An optimal value for the transmission power decreases the interference among nodes, which, in turn, increases the number of simultaneous transmissions.

• **Channel utilization:** A reduction in the transmission power increases frequency reuse, which leads to better channel reuse. Power control becomes very important for CDMA-based systems in which the available bandwidth is shared among all the users. Hence, power control is essential to maintain the required signal to interference ratio (SIR) at the receiver and to increase the channel reusability.

**4.2.3 CLASSIFICATION OF ENERGY MANAGEMENT SCHEMES**

The need for energy management in ad hoc wireless networks, discussed in the previous section, points to the fact that energy awareness needs to be adopted by the protocols at all the layers in the protocol stack, and has to be considered as one of the important design objectives for such protocols. Energy conservation can be implemented using the following techniques:

• Battery management schemes

• Transmission power management schemes

• System power management schemes

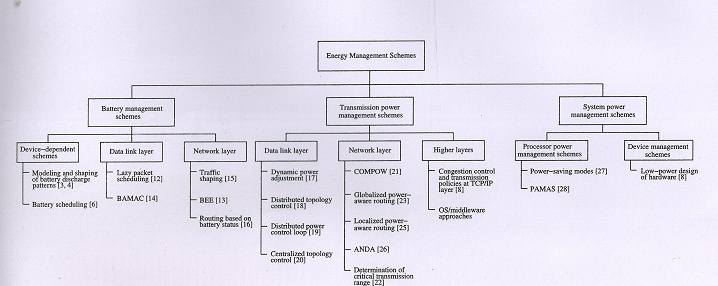
Maximizing the life of an ad hoc wireless network requires an understanding of the capabilities and the limitations of energy sources of the nodes. A greater battery capacity leads to a longer lifetime of the nodes. Increasing the capacity of the batteries can be achieved by taking into consideration either the internal characteristics of the battery (battery management) or by minimizing the activities that utilize the battery capacity (power management). The system power management approach can be further divided into the following categories:

• Device management schemes

• Processor power management schemes

Figure 4.2.1 provides an overview of some of the techniques at different layers of the protocol stack that fall into three categories: battery management, transmission power management, and system power management schemes. Though these schemes cannot be strictly classified under the different layers of the OSI protocol stack as they reside in more than one layer, the classification provided in this section is based on the highest layer in the protocol stack used by each of these protocols.

**Figure 4.2.1. Classification of energy management schemes.**



**4.2.4 BATTERY MANAGEMENT SCHEMES**

Battery-driven systems are those systems which are designed taking into consideration mainly the battery and its internal characteristics. They try to maximize the amount of energy provided by the power source by exploiting the inherent property of batteries to recover their charge when kept idle. In the following sections, we discuss the ways in which the energy efficiency of mobile wireless communication can be enhanced through the use of improved battery management techniques as described. Recent research results in this area have proved that, by varying the manner in which energy is drawn from the batteries, significant improvement can be obtained in the total amount of energy supplied by them. In the section that follows, we also discuss some of the battery characteristics which are used throughout our discussions on battery management.

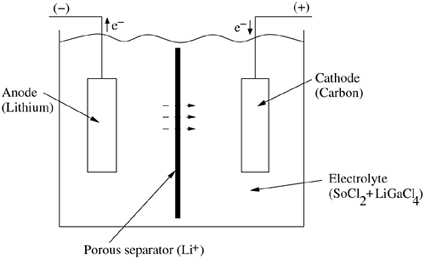
***4.2.4.1 Overview of Battery Characteristics***

The major components of batteries are illustrated in Figure 4.2.2. A battery mainly consists of an anode, a cathode, an electrolyte medium, and a case. The anode is often a metal and the cathode a metallic oxide. The electrolyte is a salt solution that promotes the ion flow. The porous separator is used to prevent a short circuit between anode and cathode by keeping them from touching one another. The battery is contained in a structural support (case) that provides dimensional stability and a positive and a negative electrode for discharging (or recharging) the cell. The positive ions move from the anode toward the cathode through the electrolyte medium and the electrons flow through the external circuit. A number of separate electrochemical cells can also be combined within the same case to create a battery.

• **Battery technologies:** The most popular rechargeable battery technologies developed over the last two decades are comprised of nickel-cadmium, lithium ion, nickel metal-hydride, reusable alkaline, and lithium polymer. The main factors considered while designing a battery technology are the energy density (the amount of energy stored per unit weight of the battery), cycle life [the number of (re)charge cycles prior to battery disposal], environmental impact, safety, cost, available supply voltage, and charge/discharge characteristics.

• **Principles of battery discharge:** A battery typically consists of an array of one or more cells. Hence, in the subsequent sections, the terms "battery" and "cell" are used interchangeably. The three main voltages that characterize a cell are: (1) the open circuit voltage (*V oc* ), that is, the initial voltage under a no-load condition of a fully charged cell, (2) the operating voltage (*V i* ), that is, the voltage under loaded conditions, and (3) the cut-off voltage (*V cut* ) at which the cell is said to be discharged. All the cells are defined by three main capacities:

**Figure 4.2.2. Basic structure of a lithium/thionyl chloride battery.**



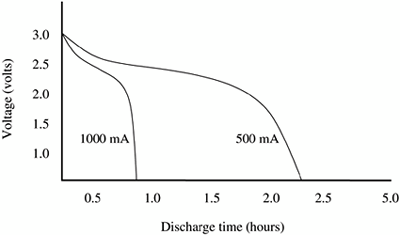
– Theoretical capacity: The amount of active materials (the materials that react chemically to produce electrical energy when the cell is discharged and restored when the cell is charged) contained in the cell refers to its theoretical capacity.

A cell cannot exceed its theoretical capacity.

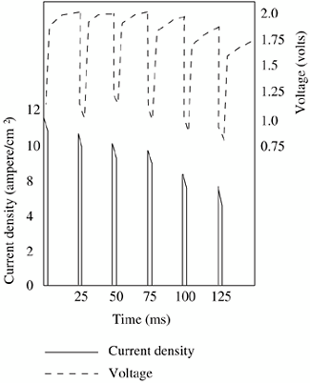
– Nominal (standard) capacity: This corresponds to the capacity actually available when discharged at a specific constant current. It is expressed in ampere-hours.

– Actual capacity: The energy delivered under a given load is said to be the actual capacity of the cell. A cell may exceed the actual capacity but not the theoretical capacity. The constant current discharge behavior of lithium-manganese dioxide (*LiMnO*2 ) cells with *V oc =* 3 *V* and *V cut =* 1 *V* is shown in Figure 4.2.3. The discharge curve is flat most of the time and a gradual slope is developed as the voltage reaches the cut-off voltage. The performance of a cell's discharge is measured using the following parameters: Discharge time: The time elapsed when a fully charged cell reaches its cut-off voltage and has to be replaced or recharged is called the discharge time of the cell. Specific power (energy): This is the power (energy) delivered by a fully charged cell under a specified discharge current. It is expressed in watt-per- kilogram (watt-hour-per-kilo gram). Discharge current: There are mainly two models of battery discharge: constant current discharge and pulsed current discharge. In pulsed current discharge, the battery switches between short discharge periods and idle periods (rest periods). After each discharge, which lasts for 3 ms, the cell was idled for 22 ms during which no recharging was allowed to take place. [Figure 4.2.4 s](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig04)hows the current density and the corresponding cell voltage. The cell is able to recover and revert to its initial open circuit voltage during the first four rest periods. After the fifth current pulse, the rest period of 22 ms turns out to be inadequate for the cell recovery.

**Figure 4.2.3. Discharge pattern of a cell when *V oc =* 3 *V* and *V cut* = 1 *V*.**



**Figure 4.2.4. Performance of a bipolar lead-acid cell subjected to six current impulses with pulse length = 3 ms and rest period = 22 ms.**



• **Impact of discharge characteristics on battery capacity:** The important chemical processes that affect the battery characteristics are given below.

– Diffusion process: When the battery is actively involved in discharging, that is, at a non-zero current, the active materials move from the electrolyte solution to the electrodes and are consumed at the electrode. If this current is above a threshold value called the *limiting current,* the active materials get depleted very quickly. But as the current decreases, the concentration of the active materials around the electrode drops. By increasing the rest time periods of the battery, longer lifetimes can be achieved due to the recovery capacity effect, which is explained later in this section. In the following discussion, we will concentrate on some of the battery management techniques which increase idle periods for batteries. Passivation process: The cell discharge is limited not only by the diffusion. process but also by a process called *passivation,* which induces in the cell the precipitation n of crystals which are produced by the discharge due to the chemical reactions on the electrode. This phenomenon increases during higher current densities. Two important effects to be considered for understanding the battery's discharge properties are stated below. Rate capacity effect: As the intensity of the discharge current increases, an insoluble component develops between the inner and outer surfaces of the cathode. The inner layer becomes inaccessible as a result of this phenomenon, rendering the cell unusable even while a sizable amount of active materials still exists. This effect depends on the actual capacity of the cell and the discharge current. Recovery capacity effect: This effect is concerned with the recovery of charges under idle conditions. By increasing the idle time, one may be able to completely utilize the theoretical capacity of the cell.

• **Battery models:** Battery models depict the characteristics of the batteries used in real life. The pros and cons of following battery models are summarized in analytical models, stochastic models, electric circuit models, and electrochemical models. Finally, a battery efficient system architecture is proposed and the following approaches are suggested to enable longer life of the nodes of an ad hoc wireless network: Supply voltage scaling: An optimal value of supply voltage (*vdd* ) is maintained, by means of scaling, that provides a balance between battery charge consumption and performance (number of packets transmitted per unit charge).

Battery-aware task scheduling: A battery-aware static scheduling scheme that optimizes the discharge power of the batteries. According to this scheme, from the knowledge of the task graph, the discharge current of the battery is shaped in order to reduce the unwanted consumption of power. This is done as a two-step process. In the first step, the initial schedule obtained is adjusted in order to reduce peak current requirements. The second step consists of a local transformation which changes the position of the scheduled events so as to minimize the delay and also the energy drawn off the cell.

Dynamic power management: Energy conservation can be achieved at the nodes carrying multimedia traffic by a graceful degradation of the quality of audio output when the battery is about to reach the completely discharged state. Many approaches have been suggested to achieve this. In one such policy, the audio device outputs high-quality sound when the remaining battery charge is above a certain threshold value. Once it falls below the threshold value, the audio device tries to degrade the output sound quality. These policies mainly exploit the recovery capacity effect of the batteries to attain theoretical capacity.

• **Battery scheduling:** The use of multiple batteries in mobile nodes has become very common. The key aspect behind this kind of an architecture is the property of charge recovery by the battery when it remains in idle state. A detailed description of the charge recovery property of the battery can be found in the next section.

• **Smart battery standard (SBS):** This is an emerging technology toward the development of batteries that consume low power. The main aim of SBS is to create standards by which the systems become aware of the batteries and interact with them in order to provide a better performance.

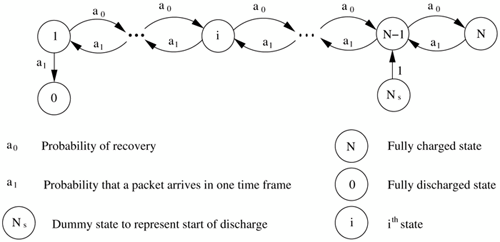
***4.2.4.2 Device-Dependent Schemes***

The lifetime of a node is determined by the capacity of its energy source and the energy required by the node. In this section, some of the device-dependent approaches that increase the battery lifetime by exploiting its internal characteristics are discussed. The stochastic model of the discharge pattern of batteries introduced in employs the following two key aspects affecting the battery life: the rate capacity effect and the recovery effect.

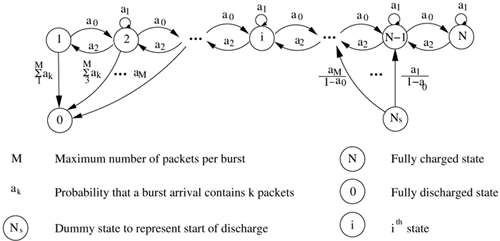
Recent works show that pulsed current discharge applied for bursty stochastic transmissions improves the battery lifetime. If pulsed current discharge is applied to a cell, significant improvement in the specific energy delivered is realized. In such an environment, higher specific power can be obtained for a constant specific energy. A model for battery pulsed discharge with recovery effect is considered. The model proposed consists of a battery with a theoretical capacity of *C* charge units and an initial battery capacity of *N* charge units. Battery behavior is considered as a discrete- time Markov process with the initial state equal to *N* and the fully discharged state 0. Time is divided into slots (frames). Each packet for the node is transmitted in one time slot and the battery enters the previous state by losing a charge unit. If the battery remains idle, it recovers one charge unit and enters the next state. The results suggest that at the most *C* (theoretical capacity) packets can be transmitted if the battery is given enough time to recover. The passivation (surface modificatio ns of metallic materials which cause an increase in their resistance to the corrosion process) time constant is assumed to be greater than the discharge time to fully drain the theoretical capacity of the cell. Thus, the passivation effects can be neglected.

In this mode, if there are packets in the queue, transmission of a packet occurs in one time slot; one charge unit is recovered if the queue is empty. The current required for transmission is drained during the entire time frame. The Markov chain for binary pulsed discharge is shown in below Figure 4.2.5. An additional dummy state is added to the Markov chain representing the cell behavior, which represents the start of the discharge. The cell is modeled as a transient process and the packet arrival follows a Bernoulli process. If the probability that a packet arrives in one time frame is stated as *a* 1 = *q* and the probability for transmitting a packet in a time slot is given by *a* 1 , then the probability of recovery is given by *a* 0 = (1 *- q*). The cell can never cross the charge state of *N.* The gain obtained in this scheme is given by , where *m p* is the total expected number of packets transmitted and *N* is the amount of charge in a fully charged battery. The gain, however, cannot exceed *C/N* where *C* is the theoretical capacity.

**Figure 4.2.5. Cell behavior under binary pulsed discharge represented using the Markov Chain Model**

**.**

**Figure 4.2.6. Cell behavior under generalized pulsed discharge represented using the Markov chain model.**



An optimal discharge strategy which provides a solution to extend the lifetime of a battery by exploiting the internal battery characteristics is proposed. The total amount of charge delivered by the battery lies between *N* and *C* units. The system is assumed to be a stochastic process with *N* states (*x*0 , ..., *xN* ). Each state *i* is denoted by the tuple (*n i , ci* ), where *n i* and *ci* are the remaining charge and capacity left in the cell, respectively. Thus the initial state is given by (*N ,C*). When the battery which is in state (*n i , ci* ) delivers *q* charge units, it moves to the state (*n i* - *q, ci* - *q*). If the battery remains idle for one charge unit, it moves to the state (*n i* + 1, *ci* ). The battery expires if either *ci* or *n i* becomes 0. When the battery is in state (*n i , ci* ), and idles for one unit of time, the probability of recovering one charge unit is given by



Here *g* is a constant value and *φ*(*ci* ) is a piecewise constant function of the number of charge units delivered which are specific to the cell's chemical properties. Using stochastic dynamic programming, an optimal policy for discharging the cell is proposed . The cells are then scheduled based on their recovery process. The efficiency of the battery is given by the equation



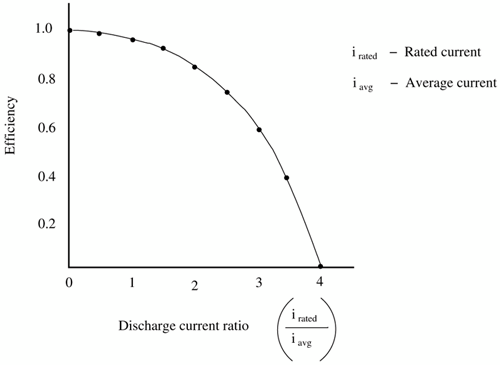
that is, the ratio of actual capacity (*E Cycle* ) to the rated capacity (*E Rated* ) which is derived from the battery specification. If the battery voltage is nearly a constant, efficiency is given by the ratio of actual to the rated current. Using this model, the battery lifetime estimation can be made as follows:

• The manufacturer specifies the rated capacity, the time constant, and the discharge plot of the battery.

• The discharge current ratio, which is the ratio between the specified rated current (*irated* ) and the calculated average current (*iavg* ), is computed.

• Efficiency is calculated by the interpolation of points in the discharge plot as seen in Figure 4.2.7, which shows the variation of efficiency with the current ratio.

**Figure 4.2.7. Variation of battery efficiency with discharge current ratio.**



Lower efficiency corresponds to a shortened battery lifetime and vice versa.

**Battery-Scheduling Techniques**

In a battery package of *L* cells, a subset of batteries can be scheduled for transmitting a given packet, leaving other cells to recover their charge. The following approaches are applied to select the subset of cells:

• **Delay-free approaches:** In the above context, *job* is defined as a demand for battery discharge which can be satisfied by the subset of cells. As soon as a job arrives, the battery charge for processing the job will be provided from the cells without any delay. The scheduling scheme for batteries can be any one of the

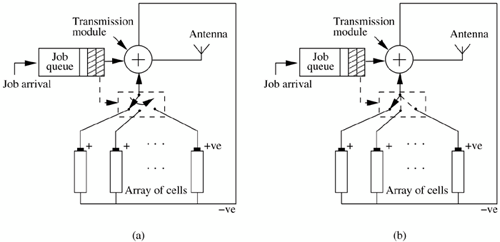
following:

**Joint technique (JN):** As soon as a job arrives, the same amount of current is drawn equally from all the cells, which are connected in parallel. If there are *L* cells, the current discharged from each of them is  times the required supply.

**Round robin technique (RR):** This scheme selects the battery in round robin

fashion and the jobs are directed to the cells by switching from one to the next one, which takes place as shown in [Figure 4.2.8 (a). T](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig08)he job from job queue gets energy from the battery selected by the transmission module based on round robin technique.

**Figure 4.2.8. Battery-scheduling techniques: (a) round robin technique (b) random technique.**



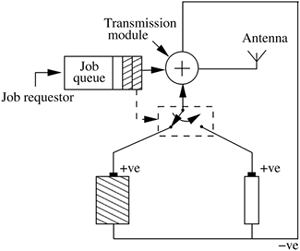
**Random technique (RN):** In this technique, any one of the cells is chosen at random with a probability of . The selected cell provides the total supply required, as shown in [Figure 4.2.8 (b).](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig08)

• **No delay-free approaches:** In these kinds of approaches, the batteries coordinate among themselves based on their remaining charge. In one such technique, a threshold is defined for the remaining charge of the cell. All the cells which have their remaining charge greater than this threshold value become eligible le for providing energy. Delay-free approaches such as round robin scheduling can be applied to these eligible le cells. The cells which are not eligible le stay in the recovery state. This enables the cells to maximize their capacity. The general battery discharge policy employed in portable devices completely drains battery packs one after the other.

**Using heterogeneous batteries:** This section examines a new model suggested for a battery-powered electronic system, which is based on the continuous time Markovian decision process (CTMDP). It attempts to exploit the two main characteristics of the rechargeable batteries, that is, the recharging capability under no load condition and the rate capacity effects discussed earlier. The main objective is to formulate an optimizatio n problem which tries to minimize the charge delivered by the battery, thereby effectively utilizing the battery capacity. The problem framed is solved using the linear programming approach. The model correlates the model of the batteries with that of the power-managed portable electronics. The model consists of two power sources which have different discharge characteristics and capacities. The jobs that arrive are serviced using the power from any of the two batteries, where the batteries are scheduled alternatively. The model, which is depicted in [Figure 4.2.9, u](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig09)ses a battery scheduler, job queue, and the job requester. Each of the three components can be modeled using the Markovian model. In this case, the battery scheduler is a simple selector that chooses between the two batteries in an alternating fashion, or it can be a scheduler that uses the round robin scheduling technique.

The formulated problem finds an optimal policy that tries to minimize the usage of the batteries that poses a constraint on the number of waiting requests in the job queue.

**Figure 4.2.9. Heterogeneous battery-scheduling technique.**



**4.2.4.3 Data Link Layer Solutions**

The data link layer solutions take into consideration battery characteristics while designing the protocols. Designing a battery-aware scheduling technique and maximizing the number of packets being transmitted are conflicting objectives. The following schemes attempt to find a trade-off between them. Subsequent sections deal with:

• Lazy packet scheduling scheme

• BAMAC protocol

Lazy Packet Scheduling Scheme

The basic principle behind the development of this scheme is that in many of the channel coding schemes for wireless transmission, the energy required to transmit a packet can be reduced significantly by minimizing the transmission power and increasing the duration of transmission. But this may not suit practical wireless environment packets. Hence, a transmission schedule is designed taking into account the delay constraints of the packets. The energy optimal offline and online scheduling algorithms consider a transmitter-receiver pair. All packets are of equal length.

Let = (*τ* 1 *, ..., τ M* ) be their transmission durations obtained as a schedule where *M* is

the number of packets to be transmitted and *w*() be the energy required to transmit a packet over the duration . Let *d* i , {*i* = 1, ..., *M*} denote the packet inter-arrival times and *k0 = 0.* Then the following parameters are defined:





For *j* ≥ 1,





**Optimal Offline Schedule**

Assuming the arrival times of all the packets (*ti* , {*i =* 1, ..., *M*}) are known *a priori* and *t*1 = 0, the problem is to find optimal values for *τ i* , 1 ≤ *i* ≤ *M*, so as to minimize . A necessary condition for optimality is



The optimal offline schedule is given by



such that, is feasible and  (time window) and satisfies the necessary condition for optimality stated above. *m j* denotes the maximum packet inter-arrival time among all the packets that arrive after the arrival of packet *j.*



**Online Algorithm**

Assuming an offline schedule as described above, the time at which packet *j* starts its transmission is given by



*b j* the backlog when the *jth* packet starts its transmission is given by



where *D i* is the inter-arrival time of *M* packets. The time *t < T* at which a packet *j* starts its transmission when there is a backlog of *b* packets can be set equal to the expected value of the random variable *E*((*b, t*)), which is evaluated numerically.



The lazy packet scheduling scheme combined with battery recovery properties is found to be providing energy saving up to 50% .

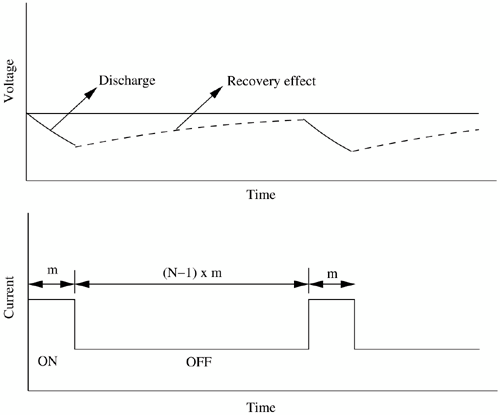
**Battery-Aware MAC Protocol**

The battery-aware MAC (BAMAC) protocol is an energy-efficient contention-based node scheduling protocol, which tries to increase the lifetime of the nodes by exploiting the *recovery capacity effect* of battery. As explained earlier in this chapter, when a battery is subjected to constant current discharge, the battery becomes unusable even while there exists a sizable amount of active materials. This is due to the rate capacity effect of the battery. If the battery remains idle for a specified time interval, it becomes possible to extend the lifetime of the battery due to the recovery capacity effect. By increasing the idle time of the battery, the whole of its theoretical capacity can be completely utilized. Also, [Equation 4.4.1 c](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11equ01)learly shows that this effect will be higher when the battery has higher remaining capacity and decreases with a decrease in the remaining battery capacity. The BAMAC protocol tries to provide enough idle time for the nodes of an ad hoc wireless network by scheduling the nodes in an appropriate manner. It tries to provide uniform discharge of the batteries of the nodes that contend for the common channel. This can be effected by using a round robin scheduling (or fair-share scheduling) of these nodes. In the BAMAC protocol, each node maintains a battery table which contains information about the remaining battery charge of each of its one-hop neighbor nodes. The entries in the table are arranged in the non-increasing order of the remaining battery charges. The RTS, CTS, Data, and ACK packets carry the remaining battery charge of the node from which they originated. A node, on listening to these packets, make a corresponding entry in its battery table. The objective of the back-off mechanism used in BAMAC protocol is to provide a near round robin scheduling of the nodes. The back-off period is given by



where, *CW min* is the minimum size of the contention window and *rank* is the position of that entry in the battery table of the node. *T SIFS* and *T DIFS* represent the SIFS and DIFS durations. Their values are same as those used in IEEE 802.11. *T t* is the is the longest possible time required to transmit a packet successfully, including the RTS-CTS-Data-ACK handshake. The node follows the back-off even for the retransmission of the packets. When this back- off scheme is followed, nodes with lesser *rank* values back off for smaller time durations compared to those with higher *rank* values. *Uniform*[0, (2*n* ×*CW min* ) - 1] returns a random number distributed uniformly in the range 0 and (2*n* × *CW min* - 1), where *n* is the number of transmission attempts made so far for a packet. Thus the nodes are scheduled based on their remaining battery capacities. The higher the battery capacity, the lower the back-off period. This ensures near round robin scheduling of the nodes. Hence, a uniform rate of battery discharge is guaranteed across all the nodes. This guarantees alternate periods of transmission and idling of the node, which leads to alternate periods of discharge and recovery of the battery, as illustrated in [Figure 4.2.10. I](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig10)n this protocol, whenever a node gains access to the channel, it is allowed to transmit only one packet, giving rise to an average idle time of (*N* - 1) × *m*, where *N* is the total number of nodes contending for the common channel and *m* is the average time taken by a node for transmission of a packet. This improves the lifetime of the battery as it gains more idle time to recover charge because of the recovery capacity effect.

**Figure 4.2.10. Illustration of BAMAC.**



**BAMAC(K) Protocol**

Unlike the BAMAC protocol wherein the nodes are allowed to transmit only one packet on gaining access to the channel, in the BAMAC(*K*) protocol proposed , *K* packets are transmitted consecutively by the node on gaining access to the channel. This provides a discharge time of *K* × *m* and an average recovery time of (*N* - 1) × *m* × *K* for the nodes. Though a larger value of *K* results in higher recovery time, it also increases the discharge time of the battery during the transmission of *K* packets. This increases the rate capacity effect due to faster depletion of the battery charge. A smaller value of *K*, on the other hand, decreases the recovery time of the battery. Hence, choosing an appropriate value for *K* is very important for optimum performance of the protocol. In the BAMAC(*K*) protocol, whenever the node attempts to gain access to the channel, it waits for DIFS time duration before transmitting the first packet. If no other neighbor transmits in this duration, the active node (the node that gains access to the channel) initiates its transmission. For transmitting each of the next *K* - 1 packets, it waits only for an SIFS duration; if the channel remains idle during this SIFS duration, the active node proceeds with the transmission of the packet. This ensures that none of the neighboring nodes gains access to the channel until the active node completes the transmission of *K* packets. This is ensured since the neighbors never find the channel idle for DIFS time duration. Both the protocols explained above ensure short-term fairness among the nodes in terms of access to the common channel. This ultimately increases the lifetime of the nodes in an ad hoc wireless network. Though the protocol provides fairness to the nodes, it does not provide per flow fairness. This is because providing per flow fairness may lead to higher battery consumption for the nodes with more flows than the nodes with lesser number of flows. Since the protocol considers improving the lifetime of the nodes as its main objective, individual flows are not taken into consideration. Another issue in this protocol lies in finding an optimal value for *K*, which depends on a number of parameters such as number of neighbor nodes contending for the channel access, packet arrival rate for all the nodes, packet deadline, traffic characteristics, and battery parameters and characteristics.

**4.2.4.4 Network Layer Solutions**

The lifetime of a network is defined as the time from which the network starts operating until the time when the first node runs out of battery charge. The network layer solutions for battery management aim mainly at increasing the lifetime of the network. The major solutions provided focus primarily on developing routing protocols that use routing metrics such as low energy cost and remaining battery charge.

**Traffic-Shaping Schemes**

This section discusses some of the traffic-shaping schemes, which are based on the battery discharge characteristics. The scheme proposed uses the same basic model for the battery explained in earlier sections and is based on the fact that most of the network traffic is bursty. Introducing some acceptable delays in the battery discharge requests paves the way for the battery to be idle for a few time slots. This allows charge recovery to a certain extent. A proper analysis of the traffic characteristic provides a discharge-shaping technique by introducing battery idle times that trade off energy efficiency and delay. We shall now discuss an algorithm that increases the lifetime of the battery by shaping the network traffic.

**Shaping Algorithm**

The main goal of the algorithm is to introduce delay slots in the battery discharge process. This is done by defining a threshold which is expressed in terms of the amount of charge. The model used in this algorithm consists of a battery with a nominal capacity of *N*, a charge request rate of *αN* , a theoretical capacity of *T*, and a threshold (expressed as a state of charge) of *B.* Whenever the state of the battery drops below *B*, it is allowed to recover through idling. The remaining requests that arrive at the system are queued up at the buffer *L* with a large buffer size to guarantee zero loss probability. As soon as the battery recovers its charge and enters state *B +* 1, it starts servicing the queued-up requests. By applying this model of shaping discharge to the cell, the gain obtained is given by . A large value for *M,* which is equal to *N* - *B*, is favorable, since it results in higher service rates and smaller average delays for the discharge requests. Performance improves as the value of *M* increases. While considering the ON-OFF process, each requiring one charge unit, the ON-OFF times are random variables based on the Pareto distribution,



Thus with an additional delay in the discharge requests, a significant improvement in the performance of the battery can be achieved. Strategies for Blocking Relay Traffic One of the main issues concerned with ad hoc wireless networks is the relay traffic. As mentioned earlier, each node deals with two kinds of traffic: relay traffic and its own traffic. A trade-off is reached between the blocking probability of the relay traffic and the battery efficiency. The intermediate nodes may not wish to transmit the whole of the neighbors' traffic. A method that calculates the optimal fraction of relay traffic. The model used has *N* number of nodes which are uniformly distributed in an ad hoc wireless network with *R*(*s, d*) set of available routes between source *s* and destination *d.* If *P*(*k, r*) is the power required to transmit from node *k* to the next node through route *r*, then the energy cost associated with route *r* is given by



Whenever a traffic session is generated at the source node *s,* a route is selected which has minimum energy cost. A relay or an intermediate node can either allow the session traffic by sending an acknowledgment to *s* or block it by sending a negative acknowledgment to *s.* If the latter is chosen, on receiving the negative acknowledgment, the source repeats the process for the next best route on the basis of energy cost. If all the routes are blocked, the session is said to be blocked. Each node tries to behave selfishly when there is relay traffic. The amount of selfishness is defined using a quantity called *sympathy*. *sympathy*(*k, r*) denotes the sympathy associated with *kth* node in route *r.* The value of *sympathy* lies between 0 and 1, which reflects the willingness of the node to accept the relay traffic. The value 0 reflects complete unwillingness and 1 reflects complete willingness of the node to accept relay traffic. It is calculated based on some of the factors affecting transmission such as energy constraints of the node and the node's location in the network. The relay node rejects relay traffic based on the total amount of data the source intends to send to the destination and the strategy used. The following two strategies are considered in order to explore the above discussed trade-off which is based on the sympathy level:

• Random strategy: Assuming a session between source *s* and destination *d,* available routes are stored in *R*(*s, d*) in the increasing order of the sympathy level. Whenever the *kth* node in route *r* receives a session request, it accepts it with a probability *sympathy*(*k, r*).

• Pay-for-it strategy: According to this strategy, each node keeps an account of the help that it had received from other nodes relaying its messages, termed *credit,* and the amount of help it has given to others by allowing the relay traffic, that is, *debit.* The node tries to help if it has received more help in the recent past and rejects if its own traffic has been rejected often, that is, the node tries to find a balance between these two parameters. The number of packets dropped by the relay nodes decreases as the number of selfish users decreases.

**Battery Energy-Efficient Routing Protocol**

The battery energy-efficient (BEE) routing protocol is an energy-effic ient routing protocol that attempts to combine the lazy packet scheduling and the traffic-shap ing schemes. The BEE routing protocol tries to find a balance in the energy consumption among all the nodes in the ad hoc wireless network. In this protocol, a new metric called *energy cost* is introduced. From the available routes, the one which has the lowest energy cost is selected. Even a route with a greater number of hops may be selected provided the route along these nodes has minimum power per link. The algorithm also insists on selecting the path with the higher battery charge in all the nodes in order to allow the recovery effect of the batteries with lesser remaining charge. The network has *K* nodes with *S* source nodes and *D* destination nodes. Any source node *s S* can transmit to the destination node *d D* through the relay nodes. The initial and instantaneous amount of battery charge are *B i* and *bi,* respectively, for any node *i.* The transmission range of all the nodes is *p.* Any node *i* is reachable to any other node *j,* only if the distance (*d ij* ) between them is less than *p.* The energy required to transmit from node *i* to node *j* is given by , only if *i* lies within the reach of *j* and vice versa. Otherwise, the energy required to transmit is infinity.





The energy required to transmit is discredited into few energy levels between *emin* and *emax*



The mean energy required for node *i* to transmit a packet is given by, where *R i* is the set of nodes whose distance from *i* is less than *p.* Whenever the number of packets transmitted by a node decreases, the energy level of the node increases significantly. This increase in energy level can be expressed as Γ(λ*i* , *ei* ), which is a function of transmission rate λ*i* and the mean energy *ei* . The new battery status (state of charge of the battery) at any time instant is given by *b i* + Γ(λ*i , ei* ) if node is idle. The energy cost function used in BEE can be defined for *kth* route  as follows:



where *s* and *d* are the source and generic destination nodes, respectively; *lij* is the link between nodes *i*and *j* of route ;  is the weighting function such that  = *A*.λ*i* with *A* being a constant; otherwise,  = 1. *P ij* denotes the energy penalty that occurs whenever a power level higher than the mean power level is required and is equal to max(0, *eij* - *ei* ).



is the minimum value of battery status among the nodes of the route . The routing protocol can be stated as follows: Whenever source *s* has data to send to destination *d,* it calculates the energy cost function of all the routes leading to the destination *d.* The best route among them is one with minimum cost function, that is, . One main disadvantage of this scheme is that the complexity of the algorithm depends on the number of routes for which the cost has to be computed. One alternative suggested to this algorithm to reduce the complexity is that the source selects a set of routes *c* on a random basis from the available list of routes to the destination. The BEE algorithm is then applied to this set alone to find the optimal route.

Energy Conserving Routing Based on Battery Status; The main goal of battery management schemes is to exploit the discharge characteristics of the battery and allow recovery. Energy-efficient routing protocols are designed to take into account this battery information in the selection of the best route.

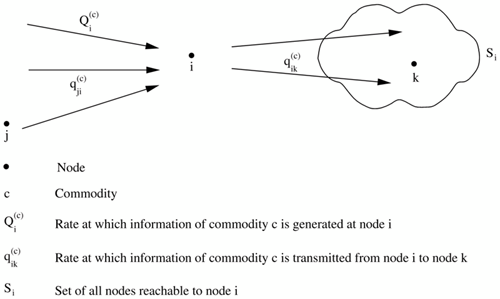
Chang and Tassiulas proposed an energy-efficient routing protocol which tries to maximize the battery lifetime. The algorithm converges to a maximizing flow problem when there is a single power level and provides an optimal lifetime for batteries. In order to maximize the lifetime, the traffic should be routed so as to balance the energy consumption among the nodes rather than trying to reduce the power consumed. Most of the previous work which deals with minimizing the overall energy consumption tries to route the packets through the path that has the minimum energy consumption per unit packet. But these routing techniques remain static, which leads to a more rapid draining of the battery charge through those routes. In Chang and Tassiulas try to find the optimal traffic split (distributing the traffic between multiple routes that exist between the source and the destination) which takes into consideration the remaining charge of the battery. The assumptions made are mentioned below. An ad hoc wireless network with *N* nodes is considered as a directed graph *G*(*N, A*) where *A* is the set of all directed links (*i, j*), where *i, j N. S i* denotes the set of all nodes reachable for node *i,* such that for any existing link (*i, j*), *j S i* . Initial battery store for node *i* is given by *E i* .  and  denote the rates at which information is generated and transmitted, respectively, at node *i* for commodity *c C,* where *C* denotes the set of all commodities. Each commodity contains a set of flows. *eij* is the energy required to transmit an information from node *i* to node *j, O*(*c*) is the set of source nodes, and *D*(*c*) is the set of destination nodes.  denotes the flow of the commodity, which is the rate at which data is transferred between node *i* and node *j.* The optimal flow condition states that the total incoming flows for node *n* must be equal to the total outgoing flows, as shown in [Figure 4.2.11.](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig11)







**Figure 4.211 Flow condition at node *i.***



The flow augmentation (FA) algorithm is performed at each node to obtain the flow which, in turn, is used to split the incoming traffic. Each node performs the following at each iteration:

• Each source node *o O*(*c*) for a commodity *c* calculates the shortest cost path to its destination *d D*(*c*).





• Then the flow is increased by an amount of , where λ is the step size.

• The shortest cost path is recalculated and the process is repeated until the first node runs out of its initial energy.

The main objective of the algorithm is to find the link with minimal cost that leads to maximization of the lifetime of the network. The three main factors that influence the cost *cij* are the energy required to transmit a packet over the link, *eij* , the initial energy *E ij* , and the instantaneous energy . A good link of flow augmenting path must consume less energy and should select the path which has more energy remaining. Simultaneous optimization of these factors is not possible. Hence, a balance point is reached between these two. Taking this into account, the cost calculation is given by



*x*1 , *x*2 , *x*3 are weighting factors for the items *eij* ,  and *E i* , respectively. The path cost is the sum of the costs of all the links on the path.

The flow redirection (FR) algorithm is an inference of the following observation, proof for which can be obtained: "In a single source, single destination environment or multiple source, multiple destination environment, without any constraints on the information generation rates, the minimal lifetime of every path remains the same under the optimal flow condition. In case of multiple source and destination, common-source and common- destination nodes are assumed with zero cost link that connects all the sources and destinations, respectively, that is, ." If there exists a flow from source *o O*(*c*) to destination  which uses the minimum total transmitted energy path with a flow value of , then the steps taken to reroute the flow to a different destination are:

• Determine the paths in which redirection is going to take place.

• Calculate the amount of redirection, that is, the percentage of flows per commodity to be redirected, given by .

• Redirect the flows through certain path to  by decrementing an amount  from the outgoing flows and by adding same amount to the flows of the

selected path.

**4.2.5 TRANSMISSION POWER MANAGEMENT SCHEMES**

The components used in the communication module consume a major portion of the energy in ad hoc wireless networks. In this section, we investigate some of the means of achieving energy conservation through efficient utilization of transmission power such as selection of an optimal power for communication. The variation in transmission power greatly influences the reach ability of a node. Increasing the transmission range not only increases coverage, but also the power consumption rate at the transmitter. This section deals with finding a trade-off between the two contradictory issues, that is, increasing the coverage of a node and decreasing its battery consumption.

***4.2.5.1 Data Link Layer Solutions***

As stated earlier, transmitter power greatly influences the reachability of the node and thus the range covered by it. Power control can be effected at the data link layer by means of topology control and constructing a power control loop. This section describes different power-based solutions at the data link layer. Some of the solutions proposed to calculate the optimum transmission range are as follows:

• Dynamic power adjustment policies

• Distributed topology control algorithms

• Constructing distributed power control loop

• Centralized topology control algorithm

Dynamic Power Adjustment Based on the Link Affinity Ad hoc wireless networks are prone to constant link failures due to node mobility, hence the stability of routes cannot be assured in such situations. But frequent link failures lead to reduced throughput. A parameter called *affinity* that decides the stability of a route is defined. Node *m* samples a set of signals from the node *n* and calculates the affinity (α*nm* )

as follows:



that is, the link is assumed to be disconnected between two nodes *n* and *m* if the signal strength *S nm* ( *current* ) is well below the threshold signal strength (*S thresh* )*.*

*Δ S nm* (*ave*) is the average of the rate of change of signal strength over the last few samples.

Each node transmits *Hello* packets to its neighbors periodically with constant power. As soon as the receiver hears one such *Hello* packet, it calculates the signal strength of the transmitter (*S t* , *t +τ*) using the relation specified below. If the time interval of the arrival of *Hello* packets is represented by *τ*, then



where *S H* is the signal strength of the *Hello* packet received, *τ* is the time period between two successive *Hello* packets, and *a* is the link affinity between a node and its neighbor. After calculating the signal strength, the node adjusts its transmission power (*P T* ) accordingly. The new adjusted transmission power (*P t* , *t+τ* ) is given by



Each node transmits with the minimum power that is required to reach the destination. Thus dynamic power adjustment can be made in a distributed way.

**Distributed Topology Control Mechanisms**

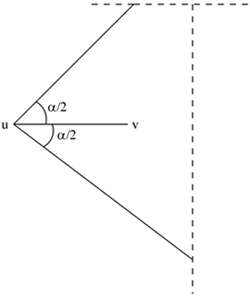
According to this algorithm, each node of the ad hoc wireless network independently runs a localized algorithm and decides the appropriate power level to be used by that node. A node increases the power directionally until it finds one node in all the directions. Then it tries to increase the lifetime of the nodes to a greater extent by reducing the transmission power and having less coverage of the nodes while guaranteeing the same connectivity as the one achieved when the nodes are maximally powered. The principle behind this algorithm is that the topology of the network can be changed by choosing the appropriate power level. An improper network topology may suffer from poor network utilization, lesser battery life, and higher delays.

The model used in this algorithm uses a cone-based topology on a two- dimensional surface. The model assumes a set *V* of *n* nodes in the plane. Each node consists of a power supply unit, memory, and processor for performing simple calculations. Any node *n* can send a broadcast message with varying powers ranging between 0 ≤ *p ≤ P.* Whenever a node *n* initiates a broadcast message, all nodes that receive the message (set *N*) reply with an acknowledgment. Thus, node *n* becomes aware of the set *N.* When any two nodes *u* and *v* exchange broadcast and acknowledgment messages, they become aware of the directions of each other which are separated by a degree of *π*. Hence, nodes *u* and *v* transmit with a power *p* and *p + π*, respectively. Techniques such as angle of arrival (AOA), which is used to calculate the direction of the node, are assumed to be available. Now we will look into the algorithm in detail, which consists of two phases.

In the first phase, which is a neighbor discovery process, a distributed algorithm is run on each node so as to form a connected network. This is done as follows. Starting with a small value for the power level, each node sends a broadcast message. Any node receiving it sends an acknowledgment back to the sender. The sender records all the acknowledgments that it received along with the information about the direction. It determines whether there exists at least one neighbor in each cone of degree *a.* Each node *u* starts with the initial value for the growing power *p.* If node *u* discovers any neighbor *v,* it adds it to the local neighbors set *N*(*u*). The node keeps increasing the power until one node is found in each cone or till the power *p* reaches the maximum value *P.* This termination condition can be mathematically formulated using Figure 4.2.12. That is, each node in the set *N*(*u*) covers a cone for any node *u.* If the union of all the cones forms an angle greater than 2*π*, the algorithm enters phase 2. The inference made from phase 1 is that, if there is a node *v* in the cone when sending with a

maximum power *P,* then there is always another node *v'* in the cone when sending with a minimum power *p*(*u*), that is, the algorithm is symmetric.

**Figure 4.2.12. Coverage determination.**



In the second phase, the redundant edges are removed without affecting the network connectivity of nodes and without removing the minimum power path. If two nodes *v* and *w* exist in the same cone for node *u,* with *v, w N*(*u*) and *w N*(*v*), then node *w* is removed from *N*(*u*) if it satisfies the condition





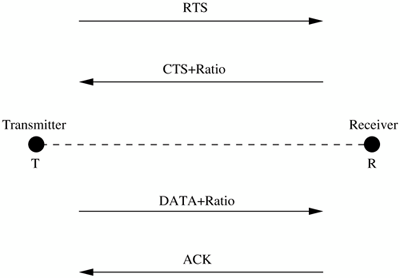


where *q ≥* 1 and *P*(*u, w*) denotes the power required by node *u* to reach *w.* For values of *P* smaller than , the algorithm guarantees maximum connected set. Constructing Distributed Power Control Loop The following is a distributed approach which tries to attain an optimal power level for the nodes in an ad hoc wireless network. The algorithm is tested on the model that assumes mobility, group communication, and fading due to blockages such as manmade obstacles. The proposed algorithm works at the MAC layer in a distributed fashion. The main objective behind the algorithm is to reduce the energy cost of communication between the nodes and thereby increasing the battery lifetime and the effective bandwidth. The main reasons behind the need for distributed algorithms in ad hoc wireless networks are the mobility and absence of a central arbiter which can inform the nodes about the power levels to be used. The algorithm aims at allowing each node to use different power levels while transmitting to different nodes and at the same time maintaining the connectivity of the network. This is because nodes that are closer require less power for transmission, The power control algorithm has been incorporated into the IEEE 802.11 MAC protocol. We now discuss the modifications made to the 802.11 MAC protocol.

• Unlike the usual IEEE 802.11 DCF protocol which uses only one common power level, the algorithm uses ten different power levels varying with a step size of one tenth of the maximum power level available.

• The format of the message header is also modified as shown in [Figure 4.2.13.](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig13) The headers of the CTS and Data frames are modified to include the information of the ratio of the signal strength of the last received message to the minimum acceptable signal strength of the node under consideration. When the receiver receives the RTS signal from the transmitter, it attaches to the CTS the ratio information calculated and sends it back to the sender. Similarly, when the sender gets the CTS packet, it includes the ratio in the Data frame and sends it to the receiver. Thus in a single transmission of RTS-CTS-Data-ACK exchange, both the sender and the receiver learn about the transmit power levels of each other.

**Figure 4.2.13. Modifications to IEEE 802.11.**



• The MAC layer for each node constructs a table which holds information about the transmit power levels of all the neighboring nodes. The information stored in the table consists of the exponential weighted average (EWA) history of the ratio for all the neighbors. This table may be small because of the fewer number of neighbors in the ad hoc network environment. There exist two situations where the table has to be considered. Whenever a message is received from a node, the receiver looks up into the EWA history. If the node is not present in the EWA table, a new entry is made in the table. The second scenario is, when the receiver is not within the range of the transmitter, that is, when there is no reply for the RTS or the Data signal, the transmitter increases its power by the step size of one tenth of the maximum power available. Similarly, the receiver increments its power level if there is no reply for the CTS sent by it. This process is continued until the node becomes reachable or the maximum power is reached. Thus each node calculates which level of power is to be used for transmission to each of its neighboring nodes.

The main aim of this power control dual channel (PCDC) protocol is to choose the transmission power in such a way that the connectivity between nodes remains unaffected but at the same time increases the number of simultaneous transmissions in the network, and thereby the throughput. The main difference between this protocol and the centralized topology control algorithm in the use of dual channels, one for transmission of control packets and the other for data packets. The use of a separate channel for control packets avoids to a great extent those collisions caused by the hidden terminal problem. This is effected as follows. Whenever a node *M* hears transmission of RTS packets in its control channel with power *p,* destined for another node *N,* which causes interference to the on-going reception at node *M,* it sends a special CTS packet which makes the RTS sender withdraw its transmission. The duration of withdrawal is specified by node *M* based on the time duration for which the on-going transmission of node *M* is estimated to last. This increases the end-to-end throughput of the system and also reduces the overall power consumption at the nodes.

**Centralized Topology Control Algorithm**

The algorithm is a centralized algorithm which adjusts the power level of the nodes to create the desired topology. The problem is constrained as an optimization problem with power level as the optimization objective and the constraints are connectivity and bi connectivity.

Unlike the conventional representation of an ad hoc wireless network using a graph, a model for the ad hoc wireless networks assumed in keeps separate the entities contributing to the ability to communicate. Some of the entities are the mobility information, propagation characteristics, and node parameters such as transmission power and antenna direction. These parameters can be defined as follows.

Any graph is said to be *k-*vertex/edge connected if and only if there are *k-* vertex/edge-d is joint paths between every pair of vertices. The graph is connected if *k =* 1 and bi-connected if *k =* 2. The network is represented as *M =* (*N, L*), where *N* is the set of nodes and *L* is the location information

,



which is the set of coordinates on the plane. The parameter vector of a node is given by *P =* {*f*0 , *f*1 , ...,*fn* }, where ƒ*i : N* → *R* is a real value where the parameter vector includes antenna configuration, spreading code, and hardware. In [[20]](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11biblio01entry20) the authors consider only one parameter, that is, power for node *u* (*p*(*u*)). Hence, *P =* {*fp*}. The propagation is represented as *γ* : *L* × *L → Z* where *L* represents the set of location coordinates in the plane and *γ*(*li , lj* ) gives the propagation loss at *lj* for the packet whose source is *li* . For successful reception



where *S* is the receiver sensitivity which is the threshold strength required for reception of signals and is assumed to be known *a priori* and *γ* is assumed to be a monotonically increasing function of the geographical distance given as



where *P* must be greater than λ(*d*) to achieve successful transmission where λ(*d*) is the *least power function* which specifies the minimum power required to transmit to a node which is at a distance *d.* Given an ad hoc wireless network represented by *M =* (*N, L*) and the transmit power function *p* and the least power function λ, the induced graph can be represented as *G=*(*V, E*), where *V* corresponds to the nodes in *N,* and *E* is the set of undirected edges such that (*u, v*) *E* if and only if both *p*(*u*) and *p*(*v*) are greater than λ(*d*(*u, v*)).



The constrained optimization problem can thus be stated as a *connected min- max power* (*CMP*) problem. The problem is to find a per-node minimal assignment of transmit powers *p : N* → *Z+*, such that the graph (*M*, λ, *L*) that is induced remains connected and the power factor *Max* *N* (*p*(*u*)) has a minimum value. For the bi-connected graph, the problem can be stated as *bi-connected augmentation min-max power* (BAMP). Given the graph as in the previous definition, the problem is to find a per-node minimal set of power increments (*δ*(*u*)) such that the induced graph (*M*, λ, *p*(*u*)*+δ*(*u*)) remains bi-connected and the power factor *Max*u *N* (*p*(*u*)*+ δ*(*u*)) has a minimum value. We shall now look into two major types of algorithms that are used to generate connected and bi-connected graphs that satisfy the given constraints. The *connect* algorithm is similar to the minimum cost spanning tree algorithm. The basic idea used in this algorithm is to iteratively merge the connected components until only one component is left. The input to the algorithm is a graph in which the nodes use the minimum power for transmission and hence remain partially connected (*M*). The following steps are performed in order to carry out this algorithm:





Step 1: First, the connected node pairs are sorted in the increasing order of the mutual distance.

Step 2: If the nodes are in different network components, the power of the nodes are increased so as to reach the other nodes.

Step 3: Step 2 is repeated until the whole network becomes connected.

The *bi-connect* algorithm attempts to discover a bi-connected graph from the given graph *M* so as to satisfy the objectives and the constraints. The extension to the bi-connected network from the algorithm *connect* can be done as follows:

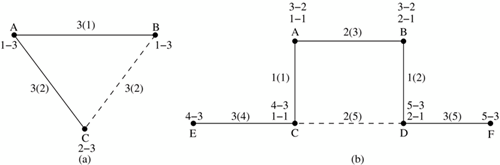
Step 1: The bi-connected components are identified in the graph induced by the algorithm *connect* based on the depth-first search method.

Step 2: The nodes are arranged in non-decreasing order of the connected node pairs as done in the previous algorithm.

Step 3: Nodes which are in different components of the network are connected by adjusting the power appropriately, and this step is repeated until the network becomes bi-connected.

Now the graph obtained may not be per-node minimal because adjusting the power to reach the nodes of other components may lead to addition of edges that are not critical. These edges are termed as *side-effect edges,* as shown in [Figure](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig14) 4.2.4. The numbers of the form *s* - *p* denote *step number* – *power* assigned during the step, *d*(*s*) denotes distance *d* between the corresponding nodes, and the step *s* during which the edge has formed. Figure 4.2.14 (a) is per-node minimal and Figure 4.2.14 (b) is obtained by reducing the power level to 1 but still maintaining the connectivity. To restore the per-node minimal for the graph shown in Figure 4.2.14 (b), a post-processing phase is carried out after applying the a fore mentioned algorithms.

**Figure 4.2.14. Side-effect edges.**



Let *S* be the list of sorted node pairs. In the *post-processing algorithm,* the

power of the nodes is reduced to the minimum possible value without affecting the connectivity of the induced graph.

In Figure 11.14 (b), Step 1 connects *A* and *C* and Step 2 connects *B* and *D* with power level 1. By increasing the power level to 2 in Step

3, *A* and *B* get connected. In Steps 4 and 5, *CE* and *DF* are formed with power

level 3. This creates a side-effect edge *CD.* Hence, the power of *A* and *B* can be reduced back to 1, without affecting the graph connectivity.

***4.2.5.2 Network Layer Solutions***

The power at the network layer can be conserved by reducing the power consumed for two main operations, namely, communication and computation. The communication-related power consumption is mainly due to the transmit- receive module present in the nodes. [Table 4.1 l](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11table01)ists the power consumption of the communication module during different modes of operation. Whenever a node remains active, that is, during transmission or reception of a packet, power gets consumed. Even when the node is not actively participating in communication, but is in the listening mode waiting for the packets, the battery keeps discharging. The computation power refers to the power spent in calculations that take place in the nodes during routing and power adjustments. The following section discusses some of the power-efficient routing algorithms. In general, a routing protocol which does not require large tables to be downloaded or greater number of calculations is preferable. Also, reducing the amount of data compression that is done before transmission may decrease the communication power but ultimately increases the number of computation tasks. Hence, a balance must be reached between the number of computation and communication tasks performed by the node, which are contradictory to each other.

**Table 4.2.1. Power consumed by Lucent ORiNOCO wireless LAN PC card in different modes**



**Common Power Protocol**

A common power protocol (COMPOW) that attempts to satisfy three major objectives: increasing the battery lifetime of all the nodes, increasing the traffic-carrying capacity of the network, and reducing the contention among the nodes.

The main reason behind the need for an optimal transmit power level for the nodes in an ad hoc wireless network is that battery power is saved by reducing the transmission range of the node. This also leads to a connected network with minimum interference.

• For the proper functioning of the RTS-CTS mechanism: If there are different power levels for each node, CTS of a lesser-powered node may not be heard by its neighboring nodes. Hence, a neighboring node may start transmitting, which leads to collision.

• For proper functioning of link-level acknowledgments : Whenever the

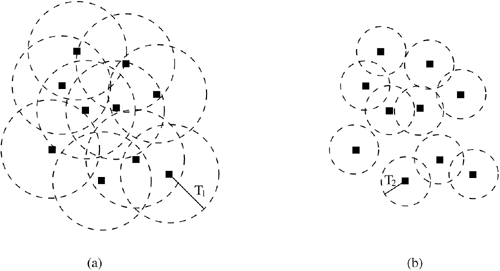
transmitter (*T*) sends a packet to a receiver (*R*), the power level at *R* must be at least equal to that of *T* so that the acknowledgment sent by node *R* reaches

node *T.* This implies that the power level of any two neighbors in an ad hoc wireless network must be equal. By transitivity, this is extended to multi-hop neighbors and thus a common power level is necessary for all the nodes in the network. If the common power level selected is a very high value, then this may lead to interference among the nodes as shown in [Figure 11.15 (a). O](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig15)n the other hand, if the value is too low, the reachability of nodes may become very weak, which in turn may render the network partitioned as shown in [Figure 4.2.15 (b).](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig15) Hence, choosing an optimum value is a difficult task. For calculating the common power level, the following solution is proposed . A network with *n* nodes is considered for study with a data rate of *W* bits/sec, in a circular area of*A* square meters. The common range to be calculated is assumed to be *r.* A successful transmission from node *T* to node *R* requires that there cannot be any other transmission occurring around a distance of (1 + Δ)*r* from *R,* where Δ > 0. Now, let us consider two simultaneous transmissions, one from *T* to *R* and another from *T'* to *R'* separated by distance of  and , respectively, as shown in [Figure 4.2.16. T](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig16)hen the distance between the receivers *R* and *R'* is given by

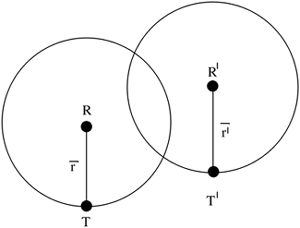


**Figure 4.2.15. Power levels and the range. (a) Interference due to higher transmission range (*T* 1 ). (b) Partition of the network due to lower**

**transmission range (*T* 2 )**



**Figure 4.2.16. Successful simultaneous transmissions.**



A conclusion drawn from the above discussion is that a circle of radius  and a circle of radius  are disjoint. Hence, the distance between any transmitter and receiver must be less than the common range *r.* If the common range for *n* nodes is *r*(*n*) then the problem can be stated as



if and only if



The maximum throughput that can be supported by the network is given by



In a practical wireless environment, factors such as the number of nodes and the area of the domain may not be known *a priori.* In such cases, rather than to deal with the range factor, it is convenient to deal directly with the power level *P.* To find the smallest power level that is required to ensure network connectivity, Proposes the following network feedback strategy. The power level for a node *j* is given by *P j* . Let *R*(*P*) denote the set of nodes which are connected when the common power level (*p* 0 =*p* 1 = ... = *p n* ) is maintained at a particular value *P*, and let *RP max* be the maximal reachable set. By analyzing the feedback and adjusting the power, the smallest power level required can be obtained. That is,



Here, *t* denotes an instant of some set of sampling times, at which the power levels are changed. *R*(*P*) can be obtained from the routing tables and hence the common power level can be calculated in real life. Kawadia and Kumar proved that the COMPOW protocol works well only in a network with a homogeneous distribution of nodes and exists only as a special case of the CLUSTERPOW protocol proposed. They have extended their COMPOW protocol to work in the presence of non- homogeneous dispersion of the nodes. This extended power control protocol, called CLUSTERPOW, is a power control clustering protocol, in which each node runs a distributed algorithm to choose the minimum power *p* to reach the destination through multip le hops. Unlike COMPOW, where all the nodes of the network agree on a common power level, in CLUSTERPOW the value of *p* can be different for different nodes and is proved to be in non-increasing sequence toward the destination. The authors in [[24] hav](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11biblio01entry24)e provided an architectural design to implement CLUSTERPOW at the network layer. This loop-free power control protocol can work in the presence of any underlying routing protocol. Globalized Power-Aware Routing Techniques

**Minimum Power Consumption Routing:**

|  |
| --- |
| The power required to transmit a packet from node *A* to node *B* is inversely proportional to the *nth*power of the distance (*d*) between them, that is, , where *n* varies from 2 to 4 depending on the distance and terrain between the nodes. A successful transmission from node *n i* to node *n j* requires the signal to noise ratio (SNR) of the node *j* to be greater than a specific threshold value This can be mathematically represented, which shows that for a successful |
| .This can be mathematically represented, which shows that for a successful |
| transmission, the SNR at receiver node*n j* given by *SNR j* must satisfy the  condition: |



where *P i* is the transmission power of host *n i* ;  is the path gain between hosts *n i* and *n j* ; *n j* is the thermal noise at the host *n j* ; and BER is the bit error rate which is based on the threshold .

The total transmission power for route *l* is the sum of the transmission powers of all nodes in the route. According to minimum power consumption routing (MPCR), the preferred route is the one with minimum total transmission power among all the availab le routes between the source and the destination. This routing algorithm can be realized by modifying the Dijkstra's shortest path algorithm. But this may select a path with a greater number of hops, which passes through nodes that require less power. Hence it may result in increasing the end-to-end delay of the packets in the network. In addition to this, involving greater number of hops may reduce the stability of the route because of the node mobility which is one of the inherent characteristics of ad hoc wireless networks. Hence, the Bellman Ford algorithm is considered, which takes into account transmission power as a cost metric. The power cost is given by



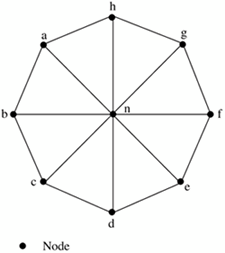
where *P transmit* (*n i, n j* ) is the transmitter power of node *i* to reach node *j* and *P transceiver* (*n j* ) is the transceiver power of node *j,* which tries to select the route with the fewer number of hops. The cost function at node *n i* is given by



where *NH*(*i*) = {*j, n j* is a neighbor node of *n i* }. This algorithm tries to reduce the overall power consumption of the network, but still lacks in the ability to reduce the power consumed at individual nodes. As shown in [Figure 4.2.17, n](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig17)ode *n* may be a common node used by multiple flows simultaneous ly. This may render

node *n* deprived of all its battery charge faster than other nodes in the network.

**Figure 4.2.17. An illustration of the disadvantage in using shortest path routing.**



Minimum Variance in Node Power Levels. The main motivation behind this metric is to ensure that all the nodes are given equal importance and no node is drained at a faster rate compared to other nodes in the network. This problem is similar to the load sharing problem in distributed systems, which is an NP-complete problem. Woo *et al.* in [[26]](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11biblio01entry26) suggest a scheme called join the shortest queue (JSQ) that tries to attain the optimal solution. According to this algorithm, for transmitting a packet, a node selects the next-hop node so that it has the least amount of traffic among all neighbors of the node. The objective can also be realized by applying a round robin selection of next-hop neighbors.

**Minimum Battery Cost Routing:**

In the minimum battery cost routing (MBCR) algorithm, individ ual battery charges are taken into consideration while selecting the route, that is, the path selected must not contain the nodes that have less remaining battery capacity. This may be done in many ways. If  denotes the battery cost at any time instant t,  represents the battery cost function of host *n i* . Now suppose the function reflects the remaining battery capacity of the node, then



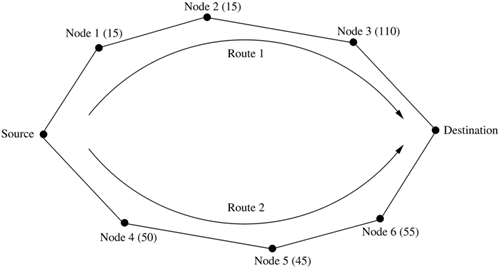
which means that higher the value of the function *fi* , the more unwilling the node is to participate in the route selection algorithm. If a route contains *N* nodes, then the total cost for the route *R i* is the sum of the cost functions of all these *N* nodes. The routing algorithm selects that path with the minimum value of the total cost among all the routes that exist between the source and the destination.



Here *A* is the set of all routes from source to destination. The major drawback with this scheme is that the use of summation of the remaining energy of the nodes as a metric selects the path which has more remaining energy on an average for all of its nodes rather than for individual nodes. This may lead to a condition as shown in [Figure 4.2.18, w](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig18)here Route 1 is selected in spite of some of its nodes having less battery capacity compared to the nodes of Route 2.

In [Figure 4.2.18, "](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig18)Node x(y)" denotes node x with y equal to the value of the function  for the node x at the time instant t. Although the battery capacity for node 3 is too little, the path containing node 3 is selected for a connection from source to destination because Route 1 has lesser battery cost due to the other nodes in the path.

**Figure 4.2.18. Illustrating the disadvantage of minimum-cost routing.**



The algorithm works well when all the nodes have higher battery capacity, but because the network nodes have almost drained their battery charges, some discharge control mechanisms are required to ensure uniform drainage from the batteries. The main advantage of this algorithm is that the metrics used can be directly incorporated in the routing protocol.

**Min-Max Battery Cost Routing**

The objective function of the min-max battery cost routing (MMBCR) algorithm is to make sure that route selection is done based on the battery capacity of all the individual nodes. Hence, the battery cost is defined as



Therefore, the desired route is given by *R i* = Min(*R j , j A*) where *A* is the set containing all possible routes. A variant of this routing algorithm minimizes the maximum cost after routing *N* packets to the destination or after a time period of *t* seconds. This tries to postpone the first node failure, which ultimately leads to a longer network lifetime. This algorithm ensures uniform discharge from the batteries. A closer look at it reveals that the path chosen does not ensure minimum transmission power and hence rapidly reduces the lifetime of all the nodes.



**Conditional Min-Max Battery Cost Routing:**

In order to solve the contradictory issues that exist in the algorithms previously mentioned, instead of using the battery cost, conditional min-max battery cost routing (CMMBCR) considers the battery capacity directly. A threshold value *γ* is defined for the remaining battery capacity of the nodes. Only those paths that have sufficiently higher capacity for all their nodes compared to the threshold participate in the routing algorithm. Once the competing routes are decided, the usual MTPR algorithm is applied on them so as to choose the path with minimum total transmission power. The battery capacity of route *j * at time *t* is



Any route *j* can participate in the routing process only if .

**Minimum Energy Disjoint Path Routing**

Srinivas and Modiano have proposed minimum energy disjoint path routing for two cases: (a) node disjoint case and (b) link disjoint case. The important need for having disjoint paths in wireless networks, especially in ad hoc wireless networks, is because of the need for reliable packet transmission and energy efficiency. Ad hoc networks are highly unreliable due to the mobility of nodes and hence the probability of link failure is quite high in such networks. This problem can be overcome by means of link disjoint routing. Also, since the ad hoc nodes have stringent battery constraints, node disjoint routing considerably increases the lifetime of the network by choosing different routes for transmitting packets at different points of time. The routing schemes assume that the topology is known *a priori* in the form of an *energy-cost* graph. The authors have proposed optimal algorithms for finding minimum energy disjoint paths and nodes in polynomial time of *O*(*kN*3) and *O*(*kN*5), where *N* is the number of nodes in the network. They have also proposed a number of sub- optimal heuristics which have a run-time of *O*(*kN*2). The authors have also proposed a distributed version of the optimal algorithms.

**Localized Power-Aware Routing Techniques**

The aim of this routing protocol is to find the shortest route to the destination so as to increase the net lifetime of the power source (battery), using localized algorithms. Local algorithms are distributed greedy algorithms that try to achieve a global objective based on the information available locally at the node. In this section, we will look into one of the power-cost-aware routing protocols proposed by Stojmenovic and Lin in [[29]](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11biblio01entry29). This protocol is based on the basic principle that, by placing intermediate nodes at the desired location between two nodes separated by a distance *d,* the transmission power can be made proportional to the distance *d* rather than *dα* [[25]](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11biblio01entry25), where *α ≥* 2. The protocol tries to find the route that minimizes the total power needed, which increases the battery lifetime. The protocol is designed to satisfy the following main objectives:

• Use of location dependent routing: The distance between the nodes is a vital piece of information that has to be taken into account to minimize the energy required for each routing task. The location of the nodes can be obtained by the methods specified below:

Using the global positioning system (GPS), the location of the nodes can be obtained by using the information obtained from the satellite.

Receiving control messages from the neighbors at regular intervals and observing the signal strengths obtained at different points of time provides the data about their distance from the node concerned.

• The routing protocol must be loop-free. This is to ensure that the path selected uses minimum power for transmission.

• When shortest path routing is applied to a graph, it may so happen that the same node is involved in several routing tasks. This eventually decreases the lifetime of that node. The protocol must distribute the traffic load so as to avoid this problem.

• The routing protocol must be designed in such a way to reduce the amount of information exchanged among the nodes, since communication incurs loss of energy. Increase in the number of communication tasks also increases the traffic in the network, which results in loss of data, retransmissions, and hence more energy consumption. The number of communication tasks can be decreased by avoiding centralized algorithms and avoiding maintenance of large routing tables.

• Selection of routes must be flexible and should try to avoid memorizing past traffic or the routes, thereby avoiding large storage.

• Another main objective is to achieve maximum packet delivery for dense networks.

• Adaptation of the routing algorithm to the continuous topological changes is important as far as ad hoc wireless networks are concerned.

The algorithm is aimed at selecting a single path for a particular routing task which guarantees delivery of packets. This is due to the mobile nature of the nodes in the ad hoc wireless networks which leads to frequent topological changes. Two nodes are said to be connected neighbors if and only if they satisfy the condition given below:



where *t*(*x*) and *d*(*A, B*) denote the transmission range of the node *x* and the distance between nodes *A* and *B*, respectively. Min-power-graphs are built using this equation. If *t*(*x*) is same for all values of *x, x* set of nodes, then the graph is said to be a unit graph. Before going into its details, we will discuss the model used in describing the properties.



• The power needed for transmitting and receiving is assumed to be *u*(*d*)= *adα* + *c* where *c* is a constant dependent on the energy spent on processing for encoding and decoding. The parameter *a* is adjusted according to the physical environment.

• When sender *S* sends packets directly to the destination *D,* let the distance between nodes *S* and *D* be given by |*SD*| = *d.* If the packets traverse through an intermediate node *A,* let |*SA| = x* and |*AD| = d* - *x.*

Let the intermediate node be placed at any arbitrary location. The following properties hold for the prescribed model:

*Lemma 1*: There always exists an intermediate node *A* between source *S* and destination *D* which reduces the energy consumption if packets from *S* destined to *D* are routed through it when the condition *d >* (*c/*(*α(*1 *-* 21-*α*)))1/*α* holds. Maximum power saving is achieved when *A* is placed exactly in the midpoint of *SD.*

*Lemma 2*: If *d >* (*c/*(*α*(1 *-* 21-*α*)))1/*α*, then by dividing *SD* into *n* equal intervals, *n* being the nearest integer to *d*(*α*(*α -* 1)*/c*1*/α*), maximum power saving can be obtained. The minimal power is then given by



First we discuss a power-saving algorithm, then a cost-saving algorithm, and finally an efficient power-cost saving algorithm derived from the previous two algorithms.

**Power-Saving Localized Routing (SP-Power) Algorithm:**

The centralized version of the above algorithm can be effected by using Dijkstra's single-source shortest weighted path algorithm, where the edge weight is *u*(*d*) = *adα + c.* This is referred to as the *SP-power algorithm.* Now the corresponding localized algorithm is as follows.

*Power Calculation*: Now we will calculate the power required to transmit a packet from node *B* (source or intermediate node) to node *D.* Let node *A* be the neighbor of *B* and let |*AB| = r, |BD| = d,* and |*AD*| = *s.* The power needed to transmit from node *B* to node *A* is *u*(*r*) = *arα+c.* Assuming that the power required for the rest of the transmissions (*v*(*s*)) in the network is uniformly distributed, by applying the above *Lemma 2* we have



The power-saving localized routing algorithm from source *S* to destination *D* is given below.

*Step 1:* Let *A* := *S*.

*Step 2:* Let *B* := *A*.

*Step 3:* Each node *B,* which may be a source or intermediate node, will select one of its neighbors *A* so as to minimize the power *p*(*B, A)=u*(*r*) *+ tv*(*s*) and sends the message to neighbor node *A.*

*Step 4:* Steps 2 and 3 are repeated until the destination is reached, that is, *A = D,* or the delivery has failed, in which case *A = B.*

**Cost-Saving Localized Routing (SP-Cost) Algorithm:**

This algorithm is based on the relation *f*(*A*) *=* 1*/g*(*A*), where *f*(*A*) is the cost of node *A,* and *g*(*A*) denotes the lifetime of the node *A,* normalized to be in the interval (0, 1). The localized version of this algorithm under constant power is mentioned below.

**Cost Calculation**: The total cost *c*(*A*) incurred when a packet moves from node *B* to node *D* via intermed iate node *A* is the sum of the node's cost *f*(*A*) =1/g(*A*) and the estimated cost of the route from node *A* to node *D.* This cost *f*(*A*) of the neighbor node *A* holding the packet is known to node *B.* The cost of the nodes in the rest of the path is proportional to the number of hops between nodes *A* and *D,* which is in turn proportional to the distance *s* = *|AD|* between nodes *A* and *D,* and inversely proportional to the transmission radius *R.* Thus the cost *f*(*A*) is *ts/R,* where *t* is a constant. Hence, the total cost can be considered to be *c*(*A*) = *ts/R + f*(*A*) or *c*(*A*) = *tsf*(*A*)*/R.* The cost-saving localized routing algorithm from source *S* to destination *D* is given below.

*Step 1:* Let *A* := *S*.

*Step 2:* Let *B* := *A.*

*Step 3:* Let each node *B*, which may be a source or intermediate node, select the neighbor node *A* that minimizes the cost *c*(*A*). If node *D* is one of the neighbors, send the message to node *D,* else send it to node *A.*

*Step 4:* Steps 2 and 3 are repeated till the destination is reached, that is, *A = D,* or the delivery has failed, in which case *A = B.*

**Power-Cost-Saving Localized Routing Algorithm**

In order to arrive at the power-cost algorithm, the power-cost of sending a message from node *B* to its neighbor node *A* must be known, which can be

power-cost(*B, A*) = *f*(*A*)*u*(*r*), where |*AB|* = *r*, or power-cost(*B, A*) = *αu*(*r*) + *βf*(*A*),

where *a* and *β* are constants. Depending on which factor is used, either sum or product, the power-cost-saving localized routing algorithm from source *S* to destination *D* can be represented as SP-Power\*Cost or SP-Power + Cost algorithm.

*Step 1:* Let *A* := *S. Step 2:* Let *B* := *A.*

*Step 3: Let each node B, which may be a source or intermediate node, select the neighbor node A that minimizes the value pc(B, A) = power-cost(B, A)+v(s)f'(A). Send the message to A.*

*Step 4:* Steps 2 and 3 are repeated till the destination is reached, that is, *A = D,* or the delivery has failed, in which case *A = B.*

**Energy-Efficie nt Ad Hoc Wireless Networks Design Algorithm:**

In ad hoc wireless networks, cluster formation is done to provide better local coordination, hierarchical addressing, and better scheduling of resources. Such cluster formations involve distributed identification of a cluster-head that has the responsibilities of internal coordination and scheduling. Inter-cluster communication is achieved through cluster gateway nodes. In a power-constrained network, the additional responsibilities deplete the energy reserve of the cluster-head faster than the other nodes in the cluster. This results in the constant change of the cluster-heads and inefficient topology. We will now discuss an energy-efficient network design algorithm called ad hoc network design algorithm (ANDA), which tries to maximize the network lifetime for a static network where cluster-heads are known *a priori.* The basic idea used in this protocol is that the cluster-heads dynamically adjust the power level; and hence, the cluster size varies based on the remaining battery charge. Thus energy is uniformly drained from the cluster-heads, which in turn increases the lifetime of the network.

Let *S C* and *S N* be the set of cluster-heads and the set of member nodes in the network, respectively. Assuming that the cluster-heads are known *a priori* and fixed

• *Covering*: This function tries to find an optimal coverage for all cluster-heads. The algorithm states that all the nodes choose one among the set of cluster- heads that projects maximum lifetime. The resulting network configuration guarantees minimum energy consumption.

• *Reconfiguration*: Finding an appropriate time for the network reconfiguration is a crucial task. It is dependent on the remaining energy of the cluster-heads. Assuming all nodes have the same energy initially and they select their cluster- heads based on the covering algorithm specified above, now after a time period *t* from the last reconfiguration, the remaining energy at the cluster-heads after a time interval *t* is given by



where *E* i (*i =* 1, ..., *C*) indicates the remaining energy at the cluster-head *i; α* and *β* are constant weighting factors; *ri* is the radius of coverage of the cluster- head; and *n i* denotes the number of nodes under the cluster-head *i.* The function covering is performed on the nodes again to assign them to the corresponding cluster-heads. Thus, a load-balancing approach of nodes is carried out in the network to increase the network lifetime.

**Determination of Critical Transmission Range:**

A centralized algorithm that calculates the minimum power level for each node that is required to maintain network connectivity based on the global information from all the nodes. This minimum value of the node's transmission range is termed the critical transmission range. This algorithm aims at finding the transmission range for each of the nodes which acts as a tradeoff between increasing the network connectivity, the spatial reusability, and the battery-life extension. The optimal value of the reception range depends on the following factors:

• Mobility: Due to the mobile nature of ad hoc wireless networks, the links are frequently formed and broken. This greatly affects the optimum range to be considered.

• Propagation models: Different fading mechanisms exist, some of which are frequency-dependent. The propagation model that results from these mechanisms has been developed for different network operational conditions such as atmospheric conditions and man-made obstacles. The use of an appropriate propagation model can help in studying the changes required in the physical layer under different environmental conditions.

• Power level: The larger the power level, the larger is the transmission range covered. This leads to faster depletion of the battery charge and hence reduces the lifetime of the node.

• Antenna configuration: Another factor to be considered is how the power is spatially distributed, which actually depends on the antenna model used. The problem is to find the critical transmission range for each of the nodes so that the network remains connected. But finding nodes within the transmission range is unpredictable due to node mobility. Some mobility model has to be assumed, to simulate the changing topology. This model can be viewed as a set of snapshots which represent the location of the mobile, separated by a small time interval. The authors prove that most of the results discussed are not sensitive to the mobility pattern under consideration. Since the topology keeps changing, maintaining the same power level for all the nodes. The network can be represented as a graph *G*(*v, e*) with *v* vertices and *e* edges. The links of a network are divided into two types: essential links which are necessary to maintain a fully connected network, and the rest of the links that form the set of redundant links. Nodes *a* and *b* are said to be direct neighbors of one another, if and only if the only closest node to *a* is *b*, and vice versa. Now the problem can be stated as

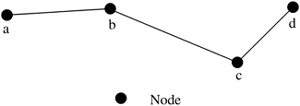


where



where *xi* denotes the location of node *i, l* is the transmission range, and *Conn*(*G*) is the connectivity of the graph, which is the ratio of the number of pairs of the nodes that are connected to the maximum number of such possible connections. It varies between 0 and 1. A link between two nodes *i* and *j* is said to be a critical link (CL) if the removal of the link leaves the network partitioned. It is the link between two *direct neighbors* that determines the critical range. Thus, the *direct neighbors* can communicate without any nodes' intervention, as shown in [Figure 4.2.19. (](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig19)*a, b*), (*b, c*), and (*c, d*) are *direct neighbors* where as (*b,d*) is not, as *c* acts as a link between them. From this, a direct neighbor graph (DNG) *G d* is created.

**Figure 4.2.19. A sample network.**





Now the problem of finding a critical link is the same as finding the longest link in DNG. On removing the loops in DNG, which can be done by deleting the links in increasing order of their lengths without affecting the network connectivity, CL can be found. The edges of DNG are a subset of *e*\*. A minimum spanning tree (MST) of the graph *G* that covers all the network nodes and aims at minimizing some link value can be used to construct DNG. This is given by



where *et* is the set of edges that provides network connectivity and represents the minimum sum of the graph edge lengths. It is also the subset of the direct neighbor set *ed*. Thus, by creating the DNG graph and applying MST algorithm for the graph, the critical link of the graph of the network can be found. This value of CL gives the critical range and hence the optimal power level for the corresponding nodes.

**4.2.5.3 Higher Layer Solutions**

This section describes some of the power-aware techniques handled at the TCP/IP and the application layers of the protocol stack. The protocols used at these layers incorporate in them power control and energy conservation.

**Congestion Control and Transmission Policies at the TCP/IP Layer:**

Due to the mobile nature of ad hoc wireless networks and the absence of the central coordinator such as a base station, the links are highly error-prone, which results in a large number of retransmissions, which in turn constantly invokes congestion control mechanisms. This kind of a situation is highly intolerable in the case of ad hoc wireless networks because of the limited energy source. Protocols at the TCP layer have to take into account the energy reserve while allowing retransmissions. Some of the suggestions are listed below.

• The number of retransmissions can be considerably reduced by making a few modifications in the error-control protocols at the TCP/IP layer, such as selective retransmissions and explicit loss notification (ELN), which try to differentiate between congestion and other types of losses.

• Error correlation greatly affects the energy conservation of ad hoc wireless networks. TCP layer protocols that take into consideration error bursts and backing off, which result in energy efficiency of the system, can be used.

• If the packets are not received, a probe cycle may be initiated by exchanging some probe messages with the transmitter rather than using congestion control algorithms directly. This achieves higher throughput while consuming less energy due to reduced traffic in the network.

**OS/Middleware and Application Layer Solutions**

A quadratic relationship exists between power and voltage. Circuit speed reduction can be done which reduces significantly the power consumed at the node. This can be compensated by implementing parallelism and pipelining. Thus, the throughput can be increased for a small value of energy consumption. There exist some sporadic events that are triggered by external events. Hence, the system can be shut down during the period of inactivity.

At the application layer, protocols such as advanced configuration and power interface (ACPI) and power management tools such as power monitor are developed to assist programmers in creating power-efficient applications. Using proxy servers results in traffic redirection or reduction in traffic, such as in multimedia where video quality may be scaled down and audio sent, and in redirecting local traffic when it arrives along with the network traffic. This results in a huge amount of energy conservation. Some energy-conserving metrics can also be used in database design such as energy per transaction. In the context of multimedia, energy conservation can be achieved by reducing the number of bits in the compressed video and by transmitting only selected information. Multimedia traffic consumes a large amount of energy in ad hoc wireless systems.

**4.2.6 SYSTEM POWER MANAGEMENT SCHEMES**

This section deals with power control in the peripherals and the processor of nodes in an ad hoc wireless network. Efficient design of the hardware brings about significant reduction in the power consumed. This can be effected by operating some of the peripheral devices in power-saving mode by turning them off under idle conditions. System power consists of the power used by all hardware units of the node. This power can be conserved significantly by applying the following schemes:

• Processor power management schemes

• Device power management schemes

***4.2.6.1 Processor Power Management Schemes***

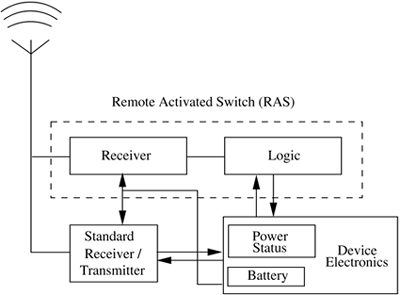
Processor power management schemes deal with techniques that try to reduce the power consumed by the processor, such as reducing the number of calculations performed. In this section, we discuss some of the power management techniques that are applied at the hardware level when there is a request from the higher layers.

**Power-Saving Modes:**

The nodes in an ad hoc wireless network consume a substantial amount of power even when they are in an idle state since they keep listening to the channel, awaiting request packets from the neighbors. In order to avoid this, the nodes are switched off during idle conditions and switched on only when there is an arrival of a request packet. This primarily has two advantages: reducing the

wastage in power consumed when the node is in the listen mode, and providing idle time for the batteries of the node to recover charges. Since the arrival of request packets is not known *a priori*, it becomes difficult to calculate the time duration for which the node has to be switched off. One solution to this problem suggested in calculates the node's switch-off time based on the quality of service (QoS) requirements. An assumption made in these systems is that the battery subsystem of the node can be remotely powered on by means of a wake- up signal that is based on RF tag technology. RF tags are used as transponders for remote localization, activation, and identification n of objects within a short range. Hard QoS requirements make the node stay active most of the time. This results in high consumption of the battery charge. A distributed technique in which a sleep pattern is selected by the power source, that is, the node selects different timeout values (duration of sleep) depending on the traffic delay and the remaining charge of the battery. The model used assumes that {1, ..., *L*} are the sleep states, with *L* corresponding to the fully active state, while 0 corresponds to the deep sleep state. The deeper the sleep state of the node, the longer is the lifetime of the battery, and the larger is the delay encountered in packet transmission or reception. To implement the remote activation of the nodes, a switch called remote activated switch (RAS) is used, as shown in [Figure 4.2.20. A](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig20)s soon as the node enters the idle state, it is switched off by the RAS switch. The receiver ofthe RAS switch still listens to the channel. It is designed to be either fully passive or powered by the battery. The remote neighbors send the wake-upsignal and a sequence. The receiver, on receiving the wake-up signal, detects the sequence. The logic circuit compares it with the standard sequence for the node. It switches on the node only if both the sequences match.

**Figure 4.2.20. Remote activated switch.**



We now discuss a power management scheme using RAS which exploits the aforementioned technique of switching off the nodes. The model used in this scheme assumes the following:

• {1, ..., *L*} are the sleep states, where *L* is the fully active state.

• Let the power consumption when the node is in one of the sleep states be denoted by *P i* (*i* = 1, ..., *L*) and the corresponding delay overhead be given

by *W i* (*i =* 1, ..., *L*-1), where *P* 1 > *P* 2 > ... > *P L* and *W 1* >*W* 2 > ... > *W L* -1 .

• Let  be the power spent on the transition from state *l* (*l*=1, ..., *L-*1) to

state *L,* and *Z l* be the minimum time that a node has to stay in a state in order to achieve a positive energy gain. The value of sleep time *Z l* in state *l* is given by



• The nodes select the sleep patterns  based on the constraint



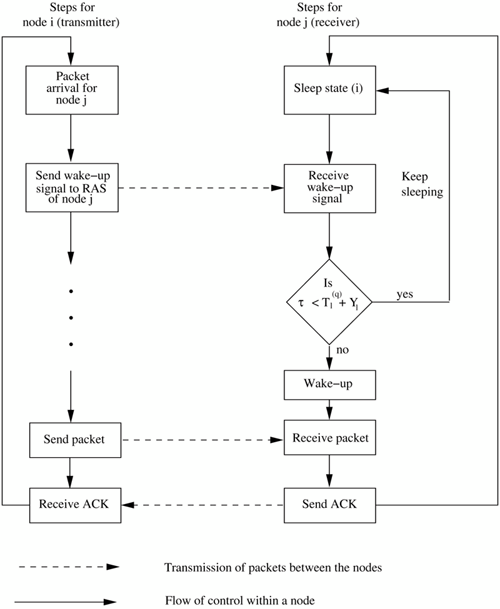
where *q =* {1, ..., *Q*}, *Yl* is the system parameter greater than *Z l* and the value of *Q* depends on the battery status and QoS requirement.

• Any node currently in state *l* wakes up if and only if



where τ *j* is the time spent by the node in state *j.* The power management scheme using RAS is shown in [Figure 4.2.21. T](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig21)he figure shows the steps followed by transmitters and receivers that use this scheme.

**Figure 4.2.21. Power management scheme using remote activated switch.**



**Power-Aware Multi-Access Signaling :**

Power-aware multi-access signaling (PAMAS) is another approach for determining the time duration for which the node should be turned off. This scheme suggests the addition of a separate signaling channel in the MACA protocol. The RTS-CTS signaling takes place in this separate channel, which determines the time period for which the node has to be powered off. The algorithm is divided into two parts:

**Addition of separate signaling channel:** This can be explained through a state diagram as shown in [Figure 4.2.22. A](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig22) node can be in any one of the six states represented within the boxes. The algorithm for the MAC layer using PAMAS is described below. Initially, when a node neither transmits nor receives packets, it stays in the *idle* state.

• *Packet transmission:*

As soon as the node gets a packet for transmission, it transmits an RTS and enters the *Await CTS* state.– If it does not receive the CTS, it enters the binary exponential back-off (*BEB*) state. A node also enters the *BEB* state if it hears a busy tone when a neighboring node which is actively transmitting sends a busy tone in the control channel.

After receiving the CTS, it enters the *Transmit packet* state and starts transmitting the packet.

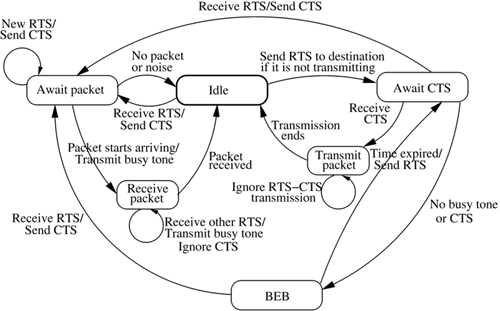
• *Packet reception:*

– As soon as a node receives an RTS, it sends a CTS back to the sender and enters the *Await packet*state, only if no other neighboring nodes exist in the *Await CTS* or *Transmit packet* state.

– If packets arrive on time, the node enters the *Receive packet* state and starts receiving the packets.

– If the packet does not arrive on time, the node enters the *idle* state again.

**Figure 4.2.22. PAMAS protocol.**



**Powering off the radios:** We now discuss the conditions under which the node enters the power-off mode:

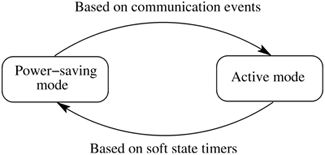
• Condition 1: The node has no packets for transmission.

• Condition 2: A neighbor node is transmitting or receiving packets, that is, the channel is busy.

The following protocol is carried out by each node to decide on the duration for which it should be powered off. When the node has packets to be transmitted, but the channel is occupied by one of its neighbors, it knows when the neighbor finishes transmission and the channel becomes free for use. If the neighbor takes *t*1 seconds to finish its transmission, then the time for which the node should be powered off is *t*1 sec. When the node powers on again and hears on-going transmission started by some other node, it again goes back to the sleep state. The time (*t*2 ) for which it remains so is based on an additional feature that has been added to the protocol which is described as follows. As soon as the node wakes up and has packets to send, it sends a probe packet *t\_probe*(*l*) to all its neighbors on the control channel, where *l* is the maximum packet length. If transmitters exist whose transmissions are expected to end in the time period (*l/2, l*), they respond to the probe packet, with a *t\_probe\_response*(*t*) packet, where *t* is the duration for which the transmission is expected to last. The node, on receiving these packets, decides on the power-off time. The *t\_probe\_response* messages may collide with other packets. In such cases, the receiver probes the channel at different intervals of time to receive the probe response packet, that is, if the node hears a collision in the interval (*t*1 , *t*2 ), it turns itself off for a time period of *t*1 . This is to enable power saving during the probing of the channel and also to ensure there are no packet losses. When the node has a packet to transmit, as soon as it powers on, it sends an RTS on the control channel. All the nodes that undergo transmission or reception send a busy tone on the control channel. If there is a collision, the node probes the channel as before. It then remains powered off for the time period min(*t, r*), where *t* and *r* are the times when the last transmitter and last receiver finished its transmission and reception, respectively. In all the above cases, if the probe message gets collided, the node stays powered on all the time.

An on-demand power management strategy in which the decision on the duration of nodes' sleep time is based on the traffic load in the network. [Figure 4.2.23 e](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig23)xplains the working of this power management strategy. As shown in the figure, every node in the ad hoc wireless network can be in either one of the two modes: power-saving mode (PSM) and active mode (AM). In the AM, the node remains awake and can transmit and receive packets. But in PSM, the node remains in the sleep state and wakes up periodically to check for messages. The transition from PSM to AM is triggered by communication events such as routing messages, and the reverse transition is triggered by the expiration of timers called keep-alive timers. The timer value is calculated based on the type of packet received recently. For example, a route update message does not set the timer value and hence does not trigger a transition of the node to active state. But an RREQ packet is used in setting the timer value. Different packet types set the timers with different values. The PSM of the neighboring nodes, which can be obtained through the *Hello* packet, also influences the timer value of a node. It has been proven that by choosing an appropriate sleep time for nodes based on the timers, a balance in the trade-off between the packet delay, energy consumption, and throughput can be attained.

**Figure 4.2.23. Illustration of on-demand power management.**



***4.2.6.2 Device Power Management Schemes***

Some of the major consumers of power in ad hoc wireless networks are the hardware devices present in the nodes. Various schemes have been proposed in the design of hardware that minimize the power consumption.

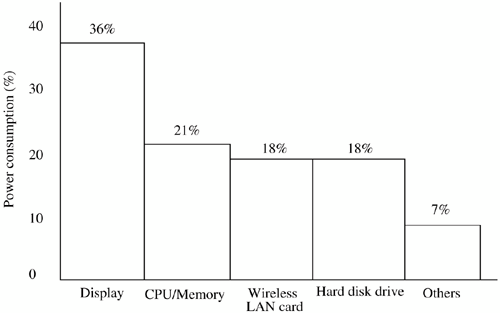
**Low-Power Design of Hardware**

Low-power design of hardware results in a significant improvement in the energy conservation. Some of the low-power design suggestions include varying clock speed CPUs, disk spin down, and flash memory. We now look into some of the sources of power consumption in the ad hoc wireless networks and the corresponding solutions to reduce power consumption.

• Major sources of power consumption in ad hoc wireless networks are the transmitters and receivers of the communication module. The design of transceivers has a significant effect on the power consumption. Hence, great care must be taken while designing them. Switching off various units of the Hardware while idling reduces the energy consumption. Instead of switching off fully, different stages may be followed in each of which there exists a different power requirement.

• The main hardware of a mobile node, in general, consists of the LCD display, DRAM, CD ROM drive, CPU, wireless interface card (in the case of a computer), and I/O subsystems. The percentage of power consumed by some of these components is shown in [Figure 4.2.24. T](https://www.safaribooksonline.com/library/view/ad-hoc-wireless/013147023X/ch11.html%23ch11fig24)he section that follows will give a brief overview of the various means of power consumption and some effective solutions suggested for low-power design of the hardware devices.

**Figure 4.2.24. Power consumed by various units of hardware.**



**CPU Power Consumption**

The energy required for the CPU operation depends largely on the clock frequency (*F*). As the clock rate increases, frequent switching of the logic gates between different voltage levels (*V*), that is, the ground voltage and the peak voltage, takes place, which leads to higher power consumption. Another effect that significantly influences the CPU power consumption is the chaining of transistors. The larger the capacitance (*C*) of these transistors, the higher the energy required. Hence, the total power required by the CPU is proportional to *CV2F.* The solution suggested is as follows:

• The parameter Ccan be set during the chip design.

• The values of *F* and *V* can be set dynamically at run-time which, along with power-aware CPU scheduling policies, reduces the power consumption significantly.

**Power-Aware CPU Scheduling:**

A small reduction in the value of the voltage *V* produces a quadratic reduction in the power consumed. This is the motivation behind the power-reduction techniques. But the voltage *V* cannot be reduced until the clock rate is reduced. Some of the approaches to reduce the clock rate given below. The CPU utilization can be balanced between peak usage and idle periods. Idle periods or sleep times can be classified in to hard or soft sleep times. Hard sleep times are unavoidable and result in compulsory idle times, whereas the soft idle times are those which can be delayed to a maximum extent possible. These two kinds of sleep times must be taken into account while balancing the CPU activities. The following algorithms suggest a few methods to calculate the CPU clock cycle time:

•OPT algorithm: This algorithm works under the ideal condition that future activities and CPU usages are known *a priori* and so the system can adjust its CPU clock rate accordingly and so maximum reduction in power consumption can be obtained by balancing the periods of activities. Though this algorithm remains unrealistic, it can be used as a benchmark for other algorithms.

• FUTURE algorithm: The basic principle behind the FUTURE algorithm is similar to that of OPT, the only difference being the window used for prediction. In the FUTURE algorithm, optimizations are made over a small window. Hence, it is realizable in a practical wireless environment.

• PAST algorithm: Rather than peering into the future, this algorithm looks into the past and decides the clock rate. It is highly unreliable because it sees the past and assumes the future with some probability. The window size on which these algorithms work acts as a trade-off between power consumption and response times.

**Hard Disk Drive (HDD) Power Consumption:**

As mentioned earlier, the basic source for power consumption in hard disks is the disk spin. Various approaches have been suggested for turning off the drives and to bring down the speed of spinning. We now see how the spin-down can be performed on the disk drives.

• By using historical data: One method suggested is based on the traces of disk usage collected over a long period of time. By analyzing various spin-down thresholds, an optimal value for threshold has to be agreed upon, which acts as a balance between the two contradictory requirements of reducing power consumption and reducing the access delays.

• Spin-up/spin-down policies: Some of the policies used in deciding the time at which the hard disk speed has to be varied are given below.

– Optimal-optimal policy: According to this policy, by having a complete knowledge of the future, the optimal values for spin-down can be obtained. This tells when the disk has to be spun down to obtain maximum efficiency, and when the disk has to be spun up to get ready for the next disk operation. This is an unrealistic policy.

– Threshold-demand policy: This algorithm forces spin-down of the disk only after attaining a certain threshold value. But the spin-up takes place only if there exists a request for the disk.

– Predictive-predictive policy: Both the spin-up and spin-down time values are predicted based on the past values.