# Routing Algorithms for Balanced Energy Consumption in Ad Hoc Networks

Hee Yong Youn
School of ECE
Sungkyunkwan University
Suwon, Korea
youn@ece.skku.ac.kr

Chansu Yu
Dept. of ECE
Cleveland State University
Cleveland, OH 44115
c.yu91@csuohio.edu

Ben Lee
Dept. of ECE
Oregon State University
Corvallis, OR 97331
benl@ece.orst.edu

## Abstract

In mobile ad-hoc network (MANET), a node communicates directly with the nodes within wireless range and indirectly with other nodes using a dynamically computed, multi-hop route via the other nodes of the MANET. In order to facilitate communication within the network, a routing protocol is used to discover routes between nodes. The primary goal of such an Ad hoc Network routing protocol is correct and efficient route establishment between a pair of nodes so that messages may be delivered in a timely manner. Although establishing efficient routes is an important goal, a more challenging goal is to provide energy efficient routing protocols, since a critical limiting factor for a mobile node is its operation time, restricted by battery capacity. However, the wireless link-only routing path in a MANET makes energy savings difficult to achieve. The corresponding reduction of nodes' lifetime directly affects the network lifetime since mobile nodes themselves collectively form a network infrastructure for routing in a MANET. This article surveys the energy aware routing mechanisms proposed for MANETs.

# 1. Introduction

Recently, wireless technology has been one of the hottest topics in computing and communications. Since the late 1970s, consumer wireless applications such as mobile phones began to take off, and presently people are beginning to activate third-generation (3G) networks for commercial purposes. Wireless networking technology offering high data rates for mobile users will flourish which will enable the handling of multimedia Web content, videoconferencing, and e-commerce, etc. *Routing* is one of the key issues for supporting these demanding applications in a rather unstable and resource limited wireless networking environment.

There are two ways to implement mobile wireless networks – *infra-structured network* and *infra-structureless (ad-hoc) network*. With an infra-structured network, mobile nodes communicate only with the base stations providing internode routing and fixed network connectivity. With the infra-structureless mobile network, each node communicates with other nodes directly or indirectly through intermediate nodes. Thus, all nodes are virtually routers participating in some protocol required for deciding and maintaining the routes.

A large number of routing protocols have been developed for *mobile ad-hoc networks* (MANETs) [14], which is characterized by unpredictable network topology changes, high-degree of mobility, energy-constrained mobile nodes, bandwidth-constrained, intermittent connection, and memory-constrained. The routing problem has been well researched in infra-structured wireless networks, where the goals are efficient detection and adaptation to the network topology, scalability, and convergence. Even though these are equally valid for MANETs, the solutions are more difficult to find since MANETs are inherently more dynamic. In particular, energy efficiency may be the most important design criteria for mobile networks since a critical limiting factor for a mobile node is its operation time, restricted by battery capacity. In infra-structured wireless networks, where a wireless link is limited to one hop between an energy-rich base station and a mobile node, the goal of energy conservation can be largely achieved by relocating power intensive network operations to the base station.

However, the wireless link-only routing path in a MANET makes energy savings difficult to achieve. The corresponding reduction of nodes' lifetime directly affects the network lifetime since mobile nodes themselves collectively form a network infrastructure for routing in a MANET. To address this problem, many research efforts have been devoted to develop energy aware network protocols such as power saving *MAC (medium access control)* layer protocols, energy efficient routing algorithms, and power sensitive network architectures. Based on the aforementioned discussion, this article focuses on the energy-aware routing mechanisms proposed for MANETs.

The remainder of the paper is organized as follows. Section 2 presents a general discussion on ad-hoc routing protocols. Although the protocols discussed in this section do not consider energy consumption as a metric for routing, they provide the basis for energy-aware routing in MANETs. Section 3 surveys the routing protocols specifically designed for balanced energy consumption in MANETs. Finally, Section 4 provides a conclusion and a discussion on power issues.

# 2. Routing Protocols for Ad Hoc Networks

The routing protocols proposed for MANETs are generally categorized as *table-driven*, *source-initiated on-demand driven*, and *hybrid* based on the timing when the routes are updated. With the table-driven routing protocols, each node attempts to maintain consistent, up-to-date routing information to every other node in the network. With source-initiated on-demand routing, route discovery and maintenance are performed only when a source node desires them. The hybrid approach combines the two approaches to minimize the overhead incurred during route discovery and maintenance. In this section, the protocols belonging to each of the three aforementioned categories are discussed.

## 2.1 Table-Driven Routing Protocols

In table-driven routing protocols, each node maintains up-to-date routing table by responding to the changes in network topology and propagating the updates. Thus, it is *proactive* in the sense that when a packet needs to be forwarded the route is already known and can be immediately used. As is the case for wired networks, each node in a MANET maintains a routing table containing a list of all the destinations, next hop, and the number of hops to each destination. The routing table is constructed using either link-state or distance vector algorithms. There are a number of protocols [23, 19, 5, 6, 7, 12, 22] that belong to this category, which are different in the number of tables manipulated for routing and the methods used for exchanging and maintaining routing tables.

Among the table-driven protocols, *Destination-Sequenced Distance Vector* (DSDV) [23], *Wireless Routing Protocol* (WRP) [19], and *Global State Routing* (GSR) [5] use destination sequence numbers to keep routes loop-free and up-to-date. These sequence numbers are assigned by the destination node and allow the mobile nodes to distinguish invalid routes from new ones. GSR is similar to the DSDV scheme, but uses the link-state instead of distance vector. Each node maintains a link-state table based on the information exchanged periodically with the neighbors. The update is selected based on the timestamp of the sequence numbers. In WRP, each node maintains a distance table, a routing table, a link-cost table and a *Message Retransmission List* (MRL) table. MRL keeps a record of which updates in an update message need to be retransmitted and which neighbors should acknowledge the retransmission [19]. An update message is sent only between neighboring nodes and contains a list of updates (the destination, the distance to the destination, and the predecessor of the destination), as well as a list of responses indicating which mobile nodes should acknowledge (ACK) the update.

In contrast to DSDV and GSR, *Cluster Gateway Switching Routing* (CGSR) [6], *Hierarchical State Routing* (HSR) [7] and *Zone-based Hierarchy Link State* (ZHLS) [12] protocols use hierarchical routing schemes. The CGSR protocol extends DSDV by grouping nodes into clusters. Thus, each cluster is represented by a *cluster-head* and two clusters can communicate via a *gateway* node that is within the communication range of the two clusters. Each node also maintains a cluster member table where the cluster heads' destinations are stored. Therefore, the cluster member table is used to perform inter-cluster routing, while the routing table is used to perform intra-cluster routing. The HSR protocol extends CGSR by forming a hierarchy of cluster heads. This is done by having nodes within a cluster broadcast their link information to each other. The cluster-head summarizes its cluster's information and sends it to neighboring cluster-heads via gateway as done in CGSR. The hierarchy reduces the overhead associated with the link-state algorithm and the number of entries in the routing table.

In ZHLS, the network is divided into non-overlapping zones without any *zone-head*. It defines two levels of topologies - node level and zone level. If any two nodes are within the communication range, a physical link exists. A virtual link exists between two zones if at least one node of a zone is physically connected to some nodes of the other zone. The node (zone) level topology provides the information on how the nodes (zones) are connected together by the physical (virtual) links. Thus, given the zone and node ID of a destination, the packet is routed based on the zone ID until it reaches the correct zone. Then, within that zone, it is routed based on node ID.

*Fisheye State Routing* (FSR) protocol [22] is another hierarchical routing scheme where information exchange is more frequent with closer nodes than that with far away nodes. FSR is an improvement over GSR to minimize the bandwidth overhead due to update messages. The FSR protocol scales well to large networks since the overhead is controlled.

## 2.2 Source-Initiated On-Demand Driven Protocols

These are reactive protocols where routes are created only when desired by the source node. The two basic procedures of source-initiated on-demand driven protocols are *route discovery* process and *route maintenance* process. The route discovery process involves sending *route-request* packets to neighbor nodes, which then forward the request to their neighbors, and so on. Once the route-request reaches the destination or the intermediate node with a “fresh enough” route, the destination/intermediate node responds by unicasting a *route-reply* packet back to the neighbor from which it first received the route-request. Once the route is established, it is maintained by some form of route maintenance process until either the destination becomes inaccessible along any path from the source or the route is no longer desired. In contrast to table-driven routing protocols, not all up-to-date routes are maintained at every node. This subsection discusses several source-initiated on-demand routing protocols [13, 24, 11, 20, 28, 8, 1]

*Dynamic Source Routing* (DSR) protocol [13] is a typical example of the on-demand protocols, where each data packet carries in its header the complete ordered list of nodes the packet passes through. This is done by having each node maintain a *route cache* that learns and caches routes to destinations. Some on-demand routing protocols are extensions of table-driven protocols. For example, the *Ad-Hoc On-Demand Vector* (AODV) protocol [24] is an improvement on the DSDV protocol, where the number of required broadcasts is minimized by creating routes on an on-demand basis. Each node maintains its own sequence number, as well as a broadcast ID for the route-request. The broadcast ID is incremented for every route-request the node initiates, and together with the node’s IP address that uniquely identifies a route-request. The *Cluster Based Routing Protocol* (CBRP) [11] is an extension of CGSR where nodes are divided into clusters. When a source has data to send, it floods route request packets only to the neighboring cluster-heads. Upon receiving the request, a cluster-head checks to see if the destination is in its cluster. If so, the request sent directly to the destination; otherwise, the request is sent to all its adjacent cluster-heads.

*Temporally Ordered Routing* (TORA) [20] is a highly adaptive protocol that provides multiple routes for any desired source-destination pair, and localizes the control messages to a very small set of nodes near the occurrence of a topological change. To accomplish this, nodes maintain routing information on adjacent (1-hop) nodes and use a “height” metric to establish a *directed acyclic graph* (DAG) rooted at the destination. When the DAG route is broken during node mobility, route maintenance is necessary to re-establish a DAG rooted at the same destination. This is achieved using a *link reversal algorithm* at the site of the link failure to re-establish the path. The algorithm tries to localize the effect and gives many alternate paths to the destination. Thus, they not only save bandwidth in updates, but also provide alternate paths on case of path failures.

In contrast to aforementioned protocols that only use the shortest path as the routing metric, *Associativity Based Routing* (ABR) [28] protocol uses the connection stability metric, called *associativity*, among mobile nodes to select the best route. In other words, a high degree of associativity may indicate a low state of node mobility, while a low degree may indicate a high state of node mobility. Associativity among nodes is determined by first having all nodes generate periodic beacons, and then the associativity tick of the receiving node with respect to the beaconing node is incremented. Thus, when packets arrive at the destination, the best route is selected by examining the associativity ticks along each of the paths. Associativity ticks are reset when the neighbors of the node or the node itself moves out of proximity.

Similarly, the *Signal Stability Routing* (SSR) Protocol [8] selects routes based on signal strength. SSR selects routes based on the signal strength between nodes and on a node's location stability, and it is divided into two cooperative protocols: The *Dynamic Routing Protocol* (DRP) and the *Static Routing Protocol* (SRP). DRP is responsible for maintaining the *Signal Stability Table* (SST) and the *Routing Table* (RT). SST records the signal strength of neighboring nodes as strong or weak using periodic beacons from each neighboring node. DRP passes a received packet to the SRP, which then forwards it using the RT. If there is no known route in RT, a route search is initiated by sending route-requests over only strong channels. The destination chooses the first arriving route-request packet to send back because it is most probable that the packet arrived over the shortest and/or least congested path. If no route-reply message is received by the source within a specific timeout period, the source node indicates that weak channels are acceptable, as these may be the only links over which the packet can be propagated.

The *Relative Distance Micro-Discovery Routing* (RDMAR) [1] protocol improves the ABR protocol by limiting the flooding of route-request packets to a certain a radius. The estimate of the radius is based on the number of radio hops between two nodes. It does not employ beaconing or a route cache.

## 2.3 Hybrid Routing Protocols

The hybrid approach combines the table-driven and source-initiated on-demand driven approaches such that the overhead incurred in route discovery and maintenance is minimized while the efficiency maximized. Several protocols belonging to this approach are presented in this subsection [10, 26, 16, 2, 17].

*Zone Routing Protocol* (ZRP) [10] partitions the network implicitly into zones, where a zone of a node includes all nearby nodes within the zone radius defined in hops. It applies proactive strategy inside the zone and reactive strategy outside the local zone. Each node may potentially be located in many zones. ZRP consists of two sub-protocols. The proactive *intra-zone routing protocol* (IARP) is an adapted distance-vector algorithm. When a source has no IARP route to a destination, it invokes a reactive *inter-zone routing protocol* (IERP), which is very similar to DSR.

The *Core Extraction Distributed Ad Hoc Routing* (CEDAR) protocol [26] is a hierarchical protocol that attempts to model the IP routing structure, with emphasis on QoS support, by identifying a subset of nodes called "*core*" nodes. Each node must be adjacent to at least one core node and picks one node as the leader or dominator. The core is determined by periodic exchange of messages between each node with its neighbors. Each core node maintains a path to the nearby nodes by issuing a limited broadcast. The core is dynamically extracted by approximating a minimum dominating set using local computation and local state, and it performs route computation on behalf of the nodes that belong to it. The bandwidth availability information is then propagated in the core subgraph. Each core node knows local links and nodes that are stable or having high bandwidth. When a source wants to send a packet to the destination, it informs its core. The core node then finds the path to the core node of the destination using some DSR-like probing. Finally, core nodes form a path using locally available link-state information.

The *Location-Aided Routing* (LAR) protocol [16] assumes that the sender has advanced knowledge of the location and velocity of the destination node using the GPS. Based on the location and velocity of the destination node, the expected zone can be defined. Thus, LAR limits the search for a new route to a small zone resulting in fewer route discovery massages. The request zone is the smallest rectangle that encompasses the expected zone. The sender explicitly specifies the request zone in its route-request message to limit the boundary on the propagation of the route-request messages.

The *Distance Routing Effect Algorithm for Mobility* (DREAM) protocol [2] uses the fact that the greater the distance separating two nodes, the slower they appear to be moving with respect to each other. Accordingly, the location information in routing tables can be updated as a function of the distance separating the nodes without compromising the routing accuracy. It sends the location updates by the moving nodes autonomously, based only on the node's mobility rate. This is because routing information on the slowly moving nodes needs to be updated less frequently than those with high mobility. This is done by sending messages in the "record direction" of the destination node, guaranteeing delivery by following the direction with a given probability.

The *Grid Location Service* (GLS) protocol [17] is a decentralized routing protocol. Each mobile node periodically updates a small set of other nodes (its *location servers*) with its current location. A node sends its position updates to its location servers without knowing their actual identities, assisted by a predefined ordering of node identifiers and a predefined hierarchy. Queries for a mobile node's location also use the predefined ordering and spatial hierarchy to find a location server for that node. For example, when node *A* wants to find the location of node *B*, it sends a request to the least node greater than or equal to node *B* for which it has location information. That node forwards the query in the same way, and so on. Eventually, the query will reach a location server of node *B*, which will then forward the query to node *B*. Since the query contains node *A*'s location, it can respond directly using geographic forwarding. Routing updates are carried out using either flooding based algorithm or link reversal algorithm.

# 3. Routing Protocols for Balanced Energy Consumption

This section surveys energy efficient routing protocols developed for MANETs. It is noted that direct comparison of these protocols is extremely difficult because these approaches have different goals with different assumptions and implementation levels. Nevertheless, there are three major issues involved in energy aware routing protocols. First, the goal is to find the path that either *minimizes* the absolute power consumed or *balances* the energy consumption of all mobile nodes. Balanced energy consumption does not necessarily lead to minimized energy consumption, but it keeps a certain node from being overloaded and thus, ensures longer network lifetime. Since the energy balance can be achieved indirectly by distributing network traffic, one such routing protocol is also discussed in this section. Second, energy awareness has been either implemented at purely routing layer or routing layer with the help from other layers such as MAC or application layer. For example, information from the MAC layer is beneficial because it usually supports power saving features which the routing protocol can exploit to provide better energy efficiency. Third, some routing protocols assume that the transmission power is controllable and nodes' location information is available (e.g., via GPS). Under these assumptions, the problem of finding a path with the least consumed power becomes a conventional optimization problem on a graph where the weighted link cost corresponds to the transmission power required for transmitting a packet between the two nodes of the link.

## 3.1 PAR (Power Aware Routing) Protocol [25]

The PAR protocol is not a new routing protocol but suggests the use of different metrics when determining a routing path. The following energy-related metrics have been suggested instead of the shortest routing path between a source and a destination: *minimizing energy consumed/packet, maximizing time to* *network partition, minimizing variance in node power levels, minimizing cost/packet,* or *minimizing maximum node cost*.

The first metric is useful for minimizing the overall energy consumption for delivering a packet. To this end, however, it is possible that some particular nodes are unfairly burdened to support many packet relaying functions. These hot spot nodes may consume more battery energy and stop running earlier than other nodes resulting in link disconnection and network partitioning. A better routing path is one where packets get routed through energy-rich intermediate nodes in spite of additional delay or hop count. Maximizing the second metric, time to network partition, is considered an ultimate goal of a MANET because it directly addresses the network lifetime. However, since it is difficult to estimate the future network behavior, the next three metrics can be used to attempt to indirectly achieve the goal. For example, the third approach, minimizing variance in node power levels, is a direct approach to maintain the energy balance with information of all nodes' power levels. In the fourth and fifth approach, each path is annotated with path cost measured by the accumulated battery life of all intermediate nodes and the minimal residual battery life among the intermediate nodes, respectively. The path with the maximum path cost is selected.

## 3.2 APR (Alternate Path Routing) Protocol [21]

The APR protocol indirectly balances energy consumption by distributing network traffic among a set of diverse paths for the same source-destination pair, called *alternate route set*. APR's performance greatly depends on the quality of the alternate route set, which can be measured by *route coupling*, i.e., how many nodes and links two routes have in common. Since the movement of a common node breaks the two routes altogether, a good alternate route set consists of decoupled routes. A decoupled alternate route set can be constructed as shown in Figure 1. When node S searches for a routing path to D, it may obtain three alternate routes: S□A□B□C□D, S□A□E□C□D, and S□E□B□D. Since they share some intermediate node(s), the alternate route set is not good enough. Each routing path is decomposed into constituent links and additional alternate routes can be constructed with improved diversity and reduced length: S□A□B□D and S□E□C□D.

Figure 1. Construction of alternate route set in the APR protocol.

With proactive routing protocols (see Subsection 2.1), each node is provided with a complete and up-to-date view of the network connectivity and thus, it is capable of identifying the best alternate routes that exist in the network. However, in the presence of significant node mobility, tracking all the changes in network connectivity can be prohibitively expensive. With reactive routing protocols (see Subsection 2.2), the alternate route set is constructed during the route discovery process since a route query may produce multiple responses containing paths to the sought-after destination. Later, during the *reply phase*, the cached path information is used to redirect replies along more diverse paths back to the source.

### 3.3 LEAR (Localized Energy Aware Routing) Protocol [29]

Compared to APR, the LEAR protocol directly controls the energy consumption. In particular, it achieves balanced energy consumption among all participating mobile nodes. The LEAR protocol is based on DSR, where the route discovery requires flooding of route-request messages. When a routing path is searched, each mobile node relies on local information of *remaining battery level* to decide whether or not to participate in the selection process of a routing path. An energy-hungry node can conserve its battery power by not forwarding data packets on behalf of others. Decision-making process in LEAR is distributed to all relevant nodes, and the destination node does not need wait or block itself in order to find the most energy efficient path.

Upon receiving a route-request message, each mobile node has the choice to determine whether or not to accept and forward the route-request message depending on its *remaining battery power* ($E_r$). When it is higher than a *threshold value ($Th_r$)*, the route-request message is forwarded; otherwise, the message is dropped. The destination will receive a route-request message only when all intermediate nodes along the route have good battery levels. Thus, the first arriving message is considered to follow an energy-efficient as well as a reasonably short path.

Table 1: The LEAR Algorithm.

| **Node** | **Steps** |
|---|---|
| Source node | Broadcast a route-request;<br>Wait for the first arriving route-reply;<br>Select the source route contained in the message;<br>Ignore all later replies; |
| Intermediate node | Upon receipt a route-request message,<br>If the message is not the first trial and $E_r < Th_r$, adjust (lower) $Th_r$ by $d$;<br>If it has the route to the destination in its cache,<br>if $E_r > Th_r$, forward (unicast) ROUTE_CACHE & ignore all later requests;<br>else, forward DROP_ROUTE_CACHE & ignore all later requests;<br>Else,<br>if $E_r > Th_r$, forward (broadcast) route-request & ignore all later requests;<br>else, forward (broadcast) DROP_ROUTE_CACHE & ignore all later requests; |
| | Upon receipt a ROUTE_CACHE,<br>If the message is not the first trial and $E_r < Th_r$, adjust (lower) $Th_r$ by $d$;<br>If $E_r > Th_r$, forward (unicast) ROUTE_CACHE & ignore all later requests;<br>else, forward (unicast) DROP_ROUTE_CACHE & ignore all later requests; and send backward (unicast) CANCEL_ROUTE_CACHE; |
| Destination node | Upon receipt the first arriving route-request or ROUTE_CACHE, send a route-reply to the source with the source route contained in the message; |

If any of the intermediate nodes along every possible path drops route-request message, the source will not receive a single reply message even though one exists. To prevent this, the source will re-send the same route-request message, but this time with an increased sequence number. When an intermediate node receives the same request message again with a larger sequence number, it adjusts (lowers) its $Th_r$ to allow forwarding to continue. Table 1 describes the LEAR algorithm. In order to reduce the repeated request messages and to utilize the route cache, four routing-related control messages are introduced: DROP_ROUTE_REQ, ROUTE_CACHE, DROP_ROUTE_CACHE and CANCEL_ROUTE_CACHE.

## 3.4 FAR (Flow Augmentation Routing) Protocol [3]

The FAR protocol maximizes network lifetime by balancing the traffic among the nodes in proportion to their energy reserves. The traffic balance, in turn, can be achieved by selecting the optimal transmission power levels and the optimal route. Given a static network topology, the selection problem turns out to be a conventional maximum flow optimization problem on a graph, where the transmission energy between two neighboring nodes corresponds to the link cost between them. Since there are multiple source-destination pairs with different data generation rates at each source, the solution can be obtained step-by-step with incremental data generation or data traffic. More specifically, FAR first solves the optimization problem with initial data traffic. It expends energy of the corresponding intermediate nodes. Then, it augments data traffic at each source and solves the same problem again with the reduced energy reserves. The final and overall routing decision is obtained by repeatedly solving the optimization problem until any node runs out of its initial energy reserves.

The cost function of the optimization problem is the sum of link cost $c_{ij}$ along the path, where $c_{ij}$ is expressed as $e_{ij}^{x1}R_i^{-x2}E_i^{x3}$, $e_{ij}$ is the energy cost for unit flow transmission over the link and $E_i$ and $R_i$ are the *initial* and *residual energy* at the transmitting node $i$, respectively. Depending on the parameters $x_1$, $x_2$, and $x_3$, the corresponding routing algorithm $FA(x_1, x_2, x_3)$ achieves different goals. In $FA(0,0,0)$, the shortest cost path is the minimum hop path and, in $FA(1,0,0)$, it is the *minimum transmitted energy (MTE) path*. $FA(1,50,50)$ in the form of $FA(1,x,x)$ balances energy consumption and significantly improves the system lifetime over the conventional MTE routing algorithm. Table 2 summarizes those routing algorithms.

Table 2: FAR routing algorithms.

| Routing Algorithm | Optimization objective |
|---|---|
| $FA(0, 0, 0)$ | Minimum hop path |
| $FA(1, 0, 0)$ | Minimum transmitted energy path |
| $FA(\square, x, x)$ | Minimum normalized residual energy used |
| $FA(\square, \square, 0)$ | Minimum absolute residual energy used |

## 3.5 OMM (Online Max-Min Routing) Protocol [18]

Data transmission sequence (or data generation rate) is not usually known in advance. Without requiring that information, the OMM protocol makes a routing decision that optimizes two different metrics: *Minimizing power consumption* and *maximizing the minimal residual power* in the nodes of the network. Given the power level information of all nodes and the power cost between two neighboring nodes, this algorithm first finds the path that minimizes the power consumption ($P_{min}$) by using the *Dijkstra algorithm*. Among the next power efficient paths with some tolerance (less than $zP_{min}$, where $z \geq 1$), it selects the best path that optimizes the second metric by iterative application of the Dijkstra algorithm with edge removals.

The parameter $z$ measures the tradeoff between the *max-min path* and the *minimum power path*. When $z$=1, the algorithm optimizes only the first metric and thus provides the minimal power consumed path. When $z$=∞, it optimizes only the second metric and thus provides the max-min path. Thus, the proper selection of the parameter $z$ is important in determining the overall performance. A perturbation method is used to compute $z$ adaptively. First, it randomly chooses an initial value of $z$, and estimates the lifetime of the most overloaded node. Then, $z$ is increased by a small constant, and the lifetime is estimated again. The two estimates are compared and the parameter $z$ is increased or decreased accordingly. Since the two successive estimates are calculated during two different time periods, the whole process is based on the assumption that the message distributions are similar as time elapses. Algorithm steps are given below.

(1) Find the path with the least power consumption, $P_{min}$, using the Dijkstra algorithm.
(2) Find the path with the least power consumption in the graph. If the power consumption > $z$ □ $P_{min}$ or no path is found, then the previous shortest path is the solution, stop.
(3) Find the minimal residual power fraction on that path, and let it be $u_{min}$.
(4) Find all the edges whose residual power fraction is smaller than $u_{min}$, remove them from graph.
(5) Go to step (2).

OMM requires information about the power levels of all mobile nodes. In large networks, this requirement is not trivial. To improve the scalability, *zone-based hierarchical routing mechanism* is used, where the area is divided into a small number of *zones*. A routing path usually consists of a global path from zone to zone and a local path (just a few hops) within the zone. With the extended OMM protocol, a node estimates the power level of each zone, computes a path across zones, and computes the best path within each zone.

## 3.6 PLR (Power-aware Localized Routing) Protocol [27]

MANET routing algorithms based on global information, such as data generation rate or power level information of other nodes, may not be practical because each node is provided with only the local information. The PLR protocol is a localized, fully distributed energy aware routing algorithm. Assuming that the location information of its neighbors and the destination is available through GPS, each node selects one of its neighbors through which the overall transmission power to the destination is minimized.

Since the transmission power needed for direct communication between two nodes has super-linear dependence on distance, it is usually energy efficient to transmit packets via intermediate nodes. For example, direct transmission from node *A* to node *D* in Figure 2 may consume more energy than indirect transmission via $N_i$ provided that $|AD|$ is larger than $(c/(a(1-2^{1-\alpha})))^{1/\alpha}$, where the transmission and reception power between two nodes separated by a distance *d* is $u(d)=ad^{\alpha}+c$. It is also shown that the power consumption is minimized, which is denoted as $v(d)$, when $(n$-1) equally spaced intermediate nodes relay transmissions along the two end nodes, where $n=d(a(\alpha-1)/c)^{1/\alpha}$ and $v(d)= dc(a(\alpha-1)/c)^{1/\alpha} + da(a(\alpha-1)/c)^{(1-\alpha)/\alpha}$.

Figure 2. Transmission from node *A* to node *D*.

Therefore, the selection of an intermediate node among its neighbors requires evaluation of *u(d)* + *v(d)*. In other words, a node (*A*), whether it is a source or an intermediate node, selects one of its neighbors ($N_1$, $N_2$, $N_3$, ...) as the next intermediate node ($N_i$) to the destination node (*D*), which minimizes $u(|AN_i|) + v(|N_iD|)$. Note that *A* to $N_i$ is a direct transmission while $N_i$ to *D* is an indirect transmission with some intermediate nodes between $N_i$ and *D*. If the goal is to maximize the network lifetime, we only need to generalize the cost function by including the remaining lifetime of node $N_i$ or all of $N_i$*'s* neighbors.

## 3.7 SPAN Protocol [4]

Unlike other aforementioned routing protocols, the SPAN operates between the routing layer and the MAC layer. This is because SPAN tries to exploit the MAC layer's power saving features in its routing decision. The basic idea of the MAC layer's power saving mechanism is to power down (*sleep*) the radio device when it has no data to transmit or receive. This allows substantial energy savings since sleep operation consumes less power. For example, Lucent's WaveLAN-II based on the IEEE 802.11 wireless LAN standard consumes 250 mA and 300 mA when receiving and transmitting, respectively, while consumes only 9 mA when it is in sleep mode [15].

In order to coordinate the sleep period operation in IEEE 802.11, one mobile node is selected as the *master*. The master node must be awake all the time and periodically sends a beacon packet to its slave nodes followed by *TIM* (*Traffic Indication Map*) that indicates the desired receivers. Each slave wakes up at the beacon times and checks if it is addressed or not. If the node is not addressed it sleeps again; otherwise, it stays awake to receive data. Figure 3 shows simple power state diagram of the IEEE 802.11 standard.

Figure 3. Power saving mechanism in IEEE 802.11 wireless LAN standard.

The SPAN protocol makes the information on master nodes available to the network layer and lets them constitute a routing backbone to route most of traffic in the MANET. All slave nodes need not wake up to forward traffic on behalf of other nodes and conserve energy by sleeping most of time. On the other hand, master nodes must be awake all the time for routing. However, this does not spend any extra energy because they need to be up anyway for MAC layer's sleep period coordination. To prevent overloading the masters and to ensure fairness, each master periodically checks if it should withdraw as a master and give other neighbors a chance to become a master.

Selecting and replacing masters must be done in a distributed way. In SPAN, each node periodically determines if it should become a master or not based on the following *master eligibility rule*: *If two of its neighbors cannot reach each other either directly or via one or two masters, it should become a master*. In Figure 4, nodes *B* and *D* become masters. Node *H* would be eligible if either *B* or *D* does not elect itself as a master yet when node *H* checks its eligibility (thus, the master selection process is not deterministic). This rule does not yield the minimum number of master nodes but it provides robust connectivity with substantial energy savings.

Figure 4. Master eligibility rule in SPAN.

## 3.8 GAF (Geographic Adaptive Fidelity) Protocol [30]

Similar to SPAN, this protocol identifies many redundant nodes with respect to routing and turns them off without sacrificing the routing fidelity. Each node uses location information based on GPS to associate itself with a "*virtual grid*", where all nodes (except master nodes) in a particular grid square are redundant with respect to forwarding packets. Thus, these nodes switch between off and listening with the guarantee that one master node in each grid stays awake to route packets. For example, in Figure 5, nodes 2, 3 and 4 in a virtual grid B are equivalent so that one of them forwards packets between nodes 1 and 5 while the other two can sleep to conserve energy. The relationship between the grid size *r* and the radio range *R* can be easily deduced as $r^2 + (2r)^2 \le R^2$ or $r \le R/\sqrt{5}$, since nodes 2 and 5 should be able to communicate directly.

Figure 5. Virtual grid structure in the GAF protocol.

Figure 6. State transition in the GAF protocol.

In GAF, nodes are in one of three states as shown in Figure 6: *sleeping*, *discovering* and *active*. Initially, a node is in the *discovery* state and exchanges discovery messages including grid IDs to find other nodes within the same grid. A node becomes *active* if it does not hear any other discovery message for $T_d$. If more than one node is in the discovery state, one with the longest expected lifetime becomes active. The active node remains active to handle routing for predefined time duration, $T_a$. After $T_a$, the node changes its state to discovery to give a chance to other nodes within the same grid to become active. In scenarios with high mobility, *sleeping* nodes should wake up earlier to take over the role of an active node, where the sleeping time $T_s$ is calculated based on the estimated time staying in the grid.

## 3.9 PEN (Prototype Embedded Network) Protocol [9]

The PEN protocol is designed for embedded networks where the rate of interaction is fairly low. It is thus more suited for control applications rather than data applications. Low power consumption is a key design criterion, which renders existing de-facto protocols replaced by low power ad hoc protocol stack from the physical layer to the transport layer. As in SPAN and GAF, this protocol exploits the low duty cycle of communication activities and powers down the radio device when it is idle. Like SPAN, the PEN system has an additional layer between the MAC and the routing layer, called the *Rendezvous layer*, which is responsible for scheduling and forecasting times of inactivity.

However, unlike SPAN, nodes interact asynchronously without master nodes and thus, costly master selection and cluster formation procedures can be avoided at the cost of extended delay. This asynchronous protocol is based on "*server beaconing*" mechanism where each node periodically wakes up, broadcasts its routing capability as a server, and listens for replies before powering down again. Any node wishing to send would wake up and listen for beacons from such nodes. Route discovery and route maintenance procedures are similar to those in AODV (See Subsection 2.2): On-demand route search and routing table exchange between neighbor nodes. Due to its asynchronous operation, the PEN protocol minimizes the amount of active time and thus saves substantial energy.

## 4. Conclusion

A MANET consists of autonomous, self-organizing and self-operating nodes. It is characterized by links with less bandwidth, nodes with energy constraints, nodes with less memory and processing power and more prone to security threats than the fixed networks. However, it has many advantages and different application areas from the fixed networks or the infra-structured mobile networks. The field of ad-hoc mobile networks is rapidly growing and changing, and while there are still many challenges that need to be met, it is likely that such networks will see wide-spread use within the next few years.

Routing is one of the main problems in MANETs. Numerous solutions to routing have been proposed, but energy efficient routing decision is more important than simple shortest path routing. In this chapter, we have provided descriptions of a number of energy aware routing schemes proposed for MANETs. While it is not clear that any particular algorithm or a class of algorithms is the best for all scenarios, each protocol has definite advantages/disadvantages and is well-suited for certain situations. Moreover, direct comparison of the energy efficient routing protocols is not possible because they are based on different assumptions such as location information availability and transmission power control. Instead, they must be carefully combined for extending the MANET lifetime.

# References

[1] Aggelou, G. and Tafazolli, R., "RDMAR: A Bandwidth-Efficient Routing Protocol for Mobile Ad Hoc Networks," ACM Int'l Workshop on Wireless Mobile Multimedia (WoWMoM), pp. 26-33, Aug. 1999.

[2] Basagni, S., Chlamtac, I., Syrotiuk, V., and Woodward, B., "A Distance Routing Effect Algorithm for Mobility (DREAM)," Int'l Conf. on Mobile Computing and Networking (MobiCom'98), pp. 76-84, 1998.

[3] Chang, J.-H., and Tassiulas, L., "Energy Conserving Routing in Wireless Ad-hoc Networks," The Conf. on Computer Communications (IEEE Infocom 2000), pp.22-31, 2000.

[4] Chen, B., Jamieson, K., Morris, R., Balakrishnan, H., "Span: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks," Int'l Conf. on Mobile Computing and Networking (MobiCom'2001), Jul. 2001.

[5] Chen, T. and Gerla, M., "Global State Routing: A new routing scheme for ad hoc wireless networks," IEEE Int'l Conf. on Communications (ICC), 1998.

[6] Chiang, C.-C., Wu, H. K., Liu, W., and Gerla, M., "Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel," IEEE SICON'97, pp. 197-211, Apr. 1997.

[7] Corson, M. and Ephremides, A., "A Distributed Routing Algorithm for Mobile Wireless Networks," ACM/Baltzer Wireless Networks Journal, Vol. 1, No.1, pp.61-81, Feb. 1995.

[8] Dube, R., Rais, C. D., Wang, K.-Y., and Tripathi, S. K., "Signal Stability based Adaptive Routing (SSA) for Ad-Hoc Networks," IEEE Personal Communications, pp. 36-45, Feb. 1997.

[9] Girling, G., Wa, J., Osborn, P., and Stefanova, R., "The Design and Implementation of a Low Power Ad Hoc Protocol Stack," IEEE Wireless Communications and Networking Conference, Sep. 2000.

[10] Haas, Z., and Pearlman, M., "Performance of query control schemes for the zone routing protocol," ACM SIGCOMM, Aug. 1998.

[11] Jiang, M., Li, J., Tay, Y. C., "Cluster Based Routing Protocol" August 1999 IETF Draft, 27 pages. http://www.ietf.org/internet-drafts/draft-ietf-manet-cbrp-spec-01.txt

[12] Joa-Ng, M., and Lu, I., "A Peer-to-Peer zone-based two-level link state routing for mobile ad hoc networks," IEEE Journal on Selected Areas in Communications, pp. 1415-1425, Aug. 1999.

[13] Johnson, D., Maltz, D., "Dynamic Source Routing in ad hoc wireless networks," Mobile Computing, edited by T. Imielinski and H. Korth, Chapter 5, pp. 153-181, Kluwer Academic Pub., 1996.

[14] Jubin, J., and Tornow, J., "The DARPA Packet Radio Network Protocols," Proceedings of the IEEE, Vol. 75, No.1, pp. 21-32, 1987.

[15] Kamerman A., and Monteban, L., "WaveLAN-II: A High-Performance Wireless LAN for the Unlicensed Band," Bell Labs Technical Journal, pp. 118-133, Summer 1997.

[16] Ko, Y. and Vaidya, N., "Location-aided routing (LAR) in mobile ad hoc networks," Int'l Conf. on Mobile Computing and Networking (MobiCom'98), pp. 66-75, 1998.

[17] Li, J., Jannotti, J., De Couto, D., Karger, D., and Morris, R., "A Scalable Location Service for Geographic Ad Hoc Routing", Int'l Conf. on Mobile Computing and Networking (MobiCom'2000), Aug. 2000.

[18] Li, Q., Aslam, J., and Rus, D., “Online Power-aware Routing in Wireless Ad-hoc Networks,” Int’l Conf. on Mobile Computing and Networking (MobiCom’2001), Jul. 2001.

[19] Murthy, S., and Garcia-Luna-Aceves, J., “An Efficient Routing Protocol for Wireless Networks," ACM Mobile Networks and Applications Journal, Special Issue on Routing in Mobile Communication Networks, pp. 183-197, Oct. 1996.

[20] Park, V., and Corson, M., “A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks,” The Conf. on Computer Communications (IEEE Infocom), Apr. 1997.

[21] Pearlman, M., Hass, Z., Sholander, P., and Tabrizi, S., “On the Impact of Alternate Path Routing for Load Balancing in Mobile Ad Hoc networks,” The First Annual Workshop on Mobile Ad Hoc Networking & Computing (MobiHOC 2000), Aug. 2000.

[22] Pei, G., Gerla, M., and Chen, T.-W., “Fisheye State Routing: A Routing Scheme for Ad Hoc Wireless Networks,” IEEE Int’l Conf. on Communications (ICC), pp. 70-74, Jun. 2000.

[23] Perkins, C., and Bhagwat, P., “Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," Computer Communications Review, pp. 234-244, Oct. 1994.

[24] Perkins, C., and Royer, E., “Ad-hoc On-Demand Distance Vector Routing," 2nd IEEE Workshop on Mobile Computing Systems and Applications, Feb. 1999.

[25] Singh, S., Woo, M., and Raghavendra, C., “Power-Aware Routing in Mobile Ad Hoc Networks,” Int’l Conf. on Mobile Computing and Networking (MobiCom’98), Oct. 1998.

[26] Sinha, P., Sivakumar, R., and Bhargavan, V., “CEMAR: a Core Extraction Distributed Ad Hoc Routing Algorithm,” The Conf. on Computer Communications (IEEE Infocom), Mar. 1999.

[27] Stojmenovic, I., and Lin, X., “Power-Aware Localized Routing in Wireless Networks,” IEEE Trans. Parallel and Distributed Systems, Vol. 12 No. 11, pp. 1122-1133, Nov. 2001.

[28] Toh, C.-K., “A Novel Distributed Routing Protocol To Support Ad-Hoc Mobile Computing,” IEEE Fifteen Annual Int’l Phonetic Conf. on Computers and Communication, pp 480-486, Mar. 1996.

[29] Woo, K., Yu, C., Youn, H. Y., and Lee, B., “Non-Blocking, Localized Routing Algorithm for Balanced Energy Consumption in Mobile Ad Hoc Networks,” Int'l Symp. on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS 2001), pp. 117-124, Aug. 2001.

[30] Xu, Y., Heidemann, J., and Estrin, D., “Geography-informed Energy Conservation for Ad Hoc Routing,” Int’l Conf. on Mobile Computing and Networking (MobiCom’2001), Jul. 2001.