Journal of Zhejiang University SCIENCE A
ISSN 1673-565X (Print); ISSN 1862-1775 (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Reliable energy-efficient routing with novel route update in wireless sensor networks

R. VIDHYAPRIYA[1], P. T. VANATHI[2]

(*[1]Department of Information Technology, PSG College of Technology, Coimbatore 641004, India*)

(*[2]Department of Electronics and Communication Engineering, PSG College of Technology, Coimbatore 641004, India*)

E-mail: vidhyar@mail.psgtech.ac.in; vanathi@ece.mail.psgtech.ac.in

Received Nov. 20, 2007; revision accepted Mar. 30, 2008

Abstract: In this paper we introduce a novel energy-aware routing protocol REPU (reliable, efficient with path update), which provides reliability and energy efficiency in data delivery. REPU utilizes the residual energy available in the nodes and the received signal strength of the nodes to identify the best possible route to the destination. Reliability is achieved by selecting a number of intermediate nodes as waypoints and the route is divided into smaller segments by the waypoints. One distinct advantage of this model is that when a node on the route moves out or fails, instead of discarding the whole original route, only the two waypoint nodes of the broken segment are used to find a new path. REPU outperforms traditional schemes by establishing an energy-efficient path and also takes care of efficient route maintenance. Simulation results show that this routing scheme achieves much higher performance than the classical routing protocols, even in the presence of high node density, and overcomes simultaneous packet forwarding.

Key words: Wireless sensor networks (WSNs), Routing, Energy efficiency, Received signal strength indicator (RSSI), Route repair, Waypoint

doi:10.1631/jzus.A072260 **Document code:** A **CLC number:** TP393

INTRODUCTION

Recent years have witnessed a growing interest in deploying a sheer number of micro-sensors that collaborate in a distributed manner on data gathering and processing. Sensors are expected to be inexpensive and can be deployed in a large scale in harsh environments, which implies that sensors are typically operating unattended. The areas of application of sensor networks vary from military use, to classroom/home use (Essa, 2000; Srivastava *et al.*, 2001), civil engineering (Tanner *et al.*, 2003; Xu *et al.*, 2004), the healthcare domain (Schwiebert *et al.*, 2001; Amato *et al.*, 2005), environmental monitoring (Jensen, 2002; Cardell-Oliver *et al.*, 2004) and the world of commerce. Examples of application include forest fire detection (Son *et al.*, 2006), habitat monitoring (Cerpa *et al.*, 2001; Mainwaring *et al.*, 2002), inventory control, energy management (Ulmer *et al.*, 2003), surveillance and reconnaissance (Delin *et al.*, 2003; Gsottberger *et al.*, 2004; Martinez *et al.*, 2004; Demirbas *et al.*, 2006).

Nodes in the sensor network have restricted storage, limited computational and energy resources. These restrictions place a limit on the types of deployable routing mechanisms. Often, sensor networks are also subject to high failure rates: connectivity between nodes can be lost due to environmental noise and obstacles; nodes may die due to battery depletion, environmental changes or malicious destruction. In such environments, reliable and energy-efficient data delivery is crucial because sensor nodes operate with limited battery power and error-prone wireless channels. Previous research on reliability only concentrates on packet retransmission, including loss detection, timer setting, acknowledgement (ACK) and negative acknowledgement (NACK) messages and error correction (Stann and Heidemann, 2003;

Mukhopadhyay *et al.*, 2004; Wan and Campbell, 2005; Zhang *et al.*, 2005). We remark that in an error-prone environment such as wireless sensor networks (WSNs) with a high packet-loss rate, the reliability goal may be reached using a retransmission mechanism, but the price is high. Retransmissions increase the traffic in the network, and thus more energy will be dissipated on communication. As discussed in many research efforts, in WSNs, battery energy is dominantly consumed by the communication functions. The more traffic also results in a higher probability of collisions and increased delivery delay. To address such issues, much research focuses on prolonging the network lifetime by exploiting energy efficiency, supporting reliability, or achieving low-cost sensor design (Al-Karaki and Kamal, 2004; Akkaya and Younis, 2005).

In this paper, an energy-aware routing with novel path update mechanism ('REPU' for short) is proposed to prolong the lifetime of the network. It achieves both reliable and energy-efficient data delivery for dense WSNs. This protocol uses the metrics 'received signal strength' (RSS) and 'available energy' to identify an energy-efficient path that minimizes packet collisions and increases the network lifetime. An energy-efficient path is formed by selecting the sensor nodes with energy above a certain threshold. In this protocol a number of intermediate nodes on a route are selected as waypoints and the route is divided into segments by the waypoints. The source and the destination are also considered as waypoints. One distinct advantage of our model is that when a node on the route moves out or fails, instead of discarding the whole original route and discovering a new route from the source to the destination, only the two waypoints of the broken segment have to find a new segment. This will have a clear performance advantage, such as low routing overhead and low end-to-end delay. This protocol can reduce packet loss due to the energy depletion of intermediate nodes. Consequently, path requests will be reduced. Thus, more energy can be used to forward data, instead of being wasted on path exploration; hence the functional life of the network is also prolonged. Simulation results show that REPU outperforms the traditional routing approaches in terms of network lifetime, packet delivery ratio and energy consumption.

The remainder of the paper is organized as follows. Section 2 provides a brief overview of the related works. Section 3 explains the operation of energy-efficient routing. Section 4 compares the performance of REPU with that of the protocols used in traditional schemes. Section 5 provides the conclusion of the work and discusses future directions.

RELATED WORKS

Sensor networks introduce new challenges that need to be dealt with as a result of their special characteristics. Their new requirements need optimized solutions at all layers of the protocol stack, by optimizing the scarce resources (Akyildiz *et al.*, 2002; Tilak *et al.*, 2002). The routing protocols of sensor networks can be classified basically into four main classes (Akkaya and Younis, 2005): data-centric, hierarchical, location-based, and network flow & QoS awareness.

(1) Data-centric algorithms are based on the use of network queries where the collected data are named to allow the nodes to search and get only the desired information. Two of the main algorithms are Directed Diffusion (Intanagonwiwat *et al.*, 2000) and SPIN (Heinzelman *et al.*, 1999).

(2) Hierarchical algorithms separate the nodes into sub-regions called 'clusters' to segregate the area of the monitoring environment. The algorithms used are LEACH (Heinzelman *et al.*, 2000), PEGASIS (Lindsey and Raghavendra, 2002) and TEEN (Manjeshwar and Agrawal, 2001). To allow communication between the clusters, a leader is selected from each cluster (cluster-head). Leaders are then responsible for the management (data aggregation, query dispatch) and transmission of the collected data in the region they control.

(3) Location-based algorithms [i.e., GAF (Xu *et al.*, 2001) and GEAR (Yu *et al.*, 2001)] rely on the use of node position information to find and forward data towards a destination in a specific network region. Position information is usually obtained from GPS (global positioning system) equipment.

(4) Network flow & QoS awareness algorithms use network traffic models and apply QoS-based mechanisms to support their routing requirements as SAR (Sohrabi *et al.*, 2000) or SPEED (He *et al.*, 2003).

In the SWR (single-path with repair routing) scheme (Tian and Georganas, 2003), a high delivery ratio is achieved by path repair whenever a break is detected. Path repair is done by using the routing information that already exists in the neighborhood of the failed node. During the path setup phase, SWR proactively attempts to establish an optimal path from each sensor node to sink node. The sensor node detects link failures by overhearing the next hop sensor node's transmission. If the sensor node does not pass the information along the path within a time period T_f, a node/channel failure is assumed. The path repair phase comes into effect. The sender broadcasts a help request, and each active neighbor checks to find an alternative path and sends back the help response message to the sender. The sender then updates its routing table and finally forwards the data packet to its new downstream node. In a further scheme (Dulaman *et al.*, 2003), the source node adds a number of redundancy bits to the original data. The resulting increased data block is fragmented and each fragment is transmitted on another path. The coding is chosen such that a subset of these fragments is sufficient to reconstruct the original data. In the routing protocol proposed in (Neha *et al.*, 2003), the traffic is spread over the nodes lying on different possible paths between the source and the sink in proposition to their residual energy. The rationale behind traffic spreading is that for a given total energy consumption in the network, every node should have spent the same amount of energy for data transmission. The objective is to assign more loads to under-utilized paths and fewer loads to over-committed paths, so that uniform resource allocation of all available paths can be ensured. Multiple paths of variable energy cost are constructed and a traffic scheduling algorithm is used to determine the order of path utilization. The multiple paths are also graded according to the QoS metric, i.e., the delay in response time. ReInForM (Deb *et al.*, 2003) is a protocol to deliver packets at a desired reliability level. It sends multiple copies of each packet along multiple paths from the source to the sink, such that data are delivered at the desired reliability.

Furthermore, most of the above work does not take into account any knowledge about the usage of the network, and hence stays completely general and detached from the application domain.

ENERGY-EFFICIENT ROUTING

A network composed of a sink node and many wireless sensor nodes in an interested area is considered. The sensor nodes are distributed randomly in the sensing field in a flat region with each node knowing its own location. The location information is usually used to determine whether (and how much) a node's sensing area overlaps with its neighbors' sensing areas, which is useful for forming the routing paths.

The sensor nodes are assumed to be fixed for their lifetimes, and the identifier of sensor nodes is determined *a priori*. Additionally, these sensor nodes have limited processing power, storage and energy, while the sink node has powerful resources to perform any tasks or to communicate with the sensor nodes. Once the nodes are deployed, they remain at their locations for sensing tasks. To allow an increase in the network lifetime, additional mechanisms are included in the routing protocol to verify other parameters set beyond the hop count that accept a more intelligent route establishment. The proposed REPU algorithm makes a decision on which neighbor a sensor node should forward the data message to. A node is selected to forward the data based on its available energy level and the RSS. Ideally, the greater the energy in the node and the farther the node from the previous one, the more likely the node to be selected as the next hop. The nodes not selected in this process will be moved to the sleep state to conserve power. The communication is assumed to be bidirectional. The potential packet collisions, which are effectively decreased by setting a random delay to every node before forwarding, are also ignored. The protocol replies with a complete route from the source node to the sink node quickly, and prepares a path that efficiently balances the energy of the nodes. REPU enables the selected nodes in the path to act as waypoint nodes to take care of route repair in case of a faulty node or a link failure.

Network setup and path discovery

The algorithm is composed of three phases: neighbor discovery, route reply and reliable transmission, by using two messages, namely 'broadcast message' and 'route reply message'. When the sink node receives an interest or an event, it launches a neighbor discovery mechanism. A broadcast message is flooded through the entire network until it reaches the source node. After the source node is reached, a route reply message is transmitted back through the ultimate neighbor to receive the request.

Messages

(1) Broadcast message: This message is transmitted when a node enters the network to execute the neighbor discovery process during the network startup and also to establish a route to the destination.

(2) Route reply: It is generated when the given source node is reached to create a new entry in the routing table.

Algorithm phases

Three phases are responsible for data forwarding in the network.

1. Neighbor discovery

Before sending the data to the sink, the sink node must start the neighbor discovery process to create a neighbor list, which is the address of all nodes that can transmit data to/from the source. During this process broadcast messages are exchanged among the nodes. The broadcast message as shown in Fig.1 consists of the source address, destination address, sequence number to distinguish the messages originating from the same source, and hop count and required energy threshold to transmit the packets and required signal strength threshold.

Source ID	Hop count	Sequence number	Required energy threshold	Required signal strength threshold	Destination ID

Fig.1 Broadcast message frame format

Unlike other energy-aware routing protocols, which attempt to find a minimum-energy-cost path, this protocol provides an energy-efficient path instead. A special flooding mechanism is adopted in the neighbor discovery. When an intermediate node receives the broadcast message, it does not forward the message to its neighbors immediately. Before sending the message out, several things are done. The intermediate node first checks its available energy. If the available energy is less than the operational energy (e.g., twice the packet transmission energy), it indicates that the node has no more energy to take more transmission jobs, and then the node simply discards the broadcast message. If the node has sufficient energy, the node measures the RSS based on the principle that a radio signal between a sender and a receiver attenuates with an increase in distance. Bahl and Padmanabhan (2000) suggest the use of average RSS to estimate distances. In general, the farther the receiving node is from the sending node, the weaker the signal is. This is true for large-scale wireless propagation models such as the free space and two-ray models (Shah and Rabaey, 2002). In small-scale propagation models such as the Rayleigh model (Ying *et al.*, 2000) and in practice, the signal strength may vary dramatically at the given radius for different directions because of obstacles. However, even in these cases, the weakening of the signal along the specific direction as the distance increases still holds. REPU protocol does not intend to precisely select the farthest node every time, but to choose nodes that are highly likely to be far away from the sender. Overall, this creates a more efficient flooding algorithm (reducing the number of retransmissions). In our simulations the signal strength threshold is fixed to be −80 dBm, since this value provides a good packet reception rate (PRR) around 85%; but for signal strength threshold values less than −85 dBm, the PRR varies rather radically. It is important to note that −85 dBm is very close to the sensitivity threshold of CC2420, which is about −90 dBm. If more than one node is within the required signal strength threshold and has sufficient energy level, and if the broadcast message is flooded again from two different nodes, message collisions will occur. In order to overcome this, the broadcast message is not rebroadcast immediately and a back-off delay scheme is applied. At the end of the current message transmission the nodes chosen to forward the broadcast message are selected by associating with a back-off timer.

Upon broadcast message reception, the receiving nodes start the timers that implement broadcast back-off delay. To ensure that optimal routes are determined, each receiving node calculates its broadcast back-off delay as a function of its distance away from the destination (this delay normally decreases along with each hop). When a node's back-off timer expires, it forwards the broadcast message and also sends an implicit acknowledgement (IACK) to the previous sender of this packet. Thus, in most cases, the node with the smallest number of hops to the destination will select itself to forward the message, simultaneously making other nodes aware of its selection. The solution described above may result in more than one node selecting itself, because not all receiving nodes may be in the broadcast range of the first selected node to overhear its IACK. To avoid such a possibility, upon receiving the IACK, the original sending node broadcasts an explicit ACK (EACK), so that all synchronized nodes know that a selection has already been made. This additional step achieves its goal under the assumption that all links of the original sending node are bidirectional. The reception of the EACK marks the end of the current self-selection round. Using the above mechanisms, the path to the source is built utilizing energy-efficient nodes. An energy-efficient node is the naturally selected node among the sender's neighbors and is usually the one with the largest available energy.

Fig.2 depicts a sample sensor network where the nodes are deployed in a random manner. All nodes within the radio range of the sink node receive the broadcast message at the same time. As shown in Fig.3, when the sink initially broadcasts the message, the nodes *A*, *E* and *G* receive the broadcast message. Assume that the available energy at *A* is larger than that at *E* and *G*, and also *A* is within the required signal strength threshold, hence node *A* is selected to forward the broadcast message to its neighboring nodes. The process continues and the selected node *B* sends out the broadcast message which is received by nodes *F* and *C*. It is found that both *F* and *C* have the same energy level and are within the required signal strength threshold. So both *F* and *C* start a back-off timer. When the back-off timer of node *F* expires, it forwards the broadcast message and sends an IACK to the previous sender of this packet, i.e., node *B*, because not all receiving nodes (node *C*) may be in

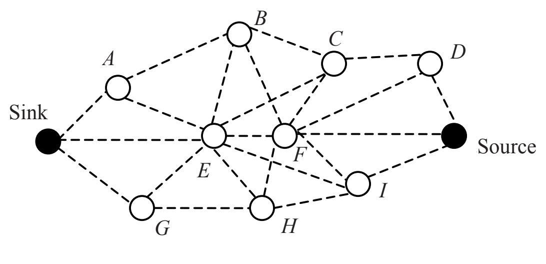

Fig.2 Network connectivity. The dashed lines indicate the radio range connectivity between two nodes

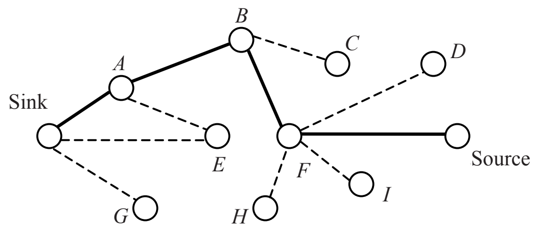

Fig.3 Path selected (the solid line) in REPU protocol

the broadcast range of the selected node (node *B*) to overhear its IACK. To avoid such a possibility, upon receiving the IACK, the original sending node *B* broadcasts an EACK, so that all nodes know that a selection has already been made. On the reception of the EACK, node *C* stops its back-off timer, and this marks the end of the current self-selection round.

When the broadcast message reaches the target 'source', the source node transmits the route reply message to the intermediate sensor node via which it received the broadcast message. Fig.4 indicates the path selected in a traditional AODV routing. It is found that the number of hops required to transmit the message in AODV is smaller than that in the energy-efficient routing protocol. But regarding the network lifetime, the energy-efficient routing protocol withstands for a longer time than a traditional AODV routing protocol.

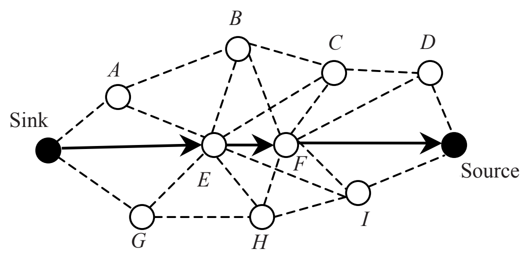

Fig.4 Path selected (the solid line with arrow) using AODV protocol

2. Selection of waypoints

The path between the sink and the source is established by the parameters 'energy threshold' and 'signal strength threshold'. A parameter called 'SEGMENT_LENGTH' (l_s) is defined in our simulations to indicate the number of waypoint nodes to be selected (by default l_s is set to 2). There can be a change in this value to make it adaptive. When the source node receives the broadcast message, based on the length of the path recorded in the hop count field of the broadcast message, the source node divides the path into segments by selecting waypoint nodes from the path. It is worth noting that there are several ways to select waypoint nodes. A novel method in selecting waypoint nodes is to divide the path into segments evenly, i.e., the length (hop count) of each segment is roughly equal to l_s. For example, suppose the path from a sink *A* to a source *J* is *A*-*B*-*C*-*D*-*E*-*F*-*G*-*H*-*I*-*J* and l_s=3, then nodes *A*, *D*, *G* and *J* are selected as the waypoint nodes.

Fig.5 gives an example of how a route from a sink node to a source node is divided into segments. The nodes 'sink', 'source' and *B* are selected as waypoint nodes, which divide the route into two segments: 'sink'-*A*-*B* and *B*-*F*-'source'. The other nodes between the waypoint nodes are forwarding nodes.

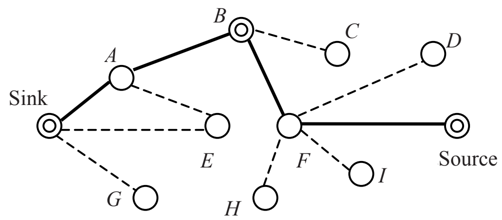

Fig.5 Selection of waypoint nodes. The double circle in the solid line indicates the selected waypoint node

3. Route reply

The destination node, upon receiving the broadcast message, will reply with a route reply message. Unlike the broadcast message, the route reply message does not rely on flooding to find its return path back to the sink; it just uses the sensor nodes through which it received the broadcast message. There are two tables maintained by all nodes in the network: waypoint routing table and forwarder routing table. The waypoint routing table maintains a route from the current node to the source node which only contains a list of waypoint nodes. Therefore, any two neighboring nodes in the waypoint routing table are one segment away from each other, instead of one hop away. The forwarder routing table maintains a list of all the selected nodes in the path other than waypoint nodes. Typically, one segment contains multiple hops. An entry in the forwarder routing table uses the 'waypoint ID: next hop node ID' structure. It stores the next hop node ID which is used to reach the waypoint node of the current segment.

When an intermediate node receives the route reply message, the node determines whether the current node is a waypoint node or a forwarding node on the route. If the node is a waypoint node, both waypoint routing table and forwarder routing table are updated; if the node is a forwarding node, it updates only its routing table. When the route reply message reaches the sink node, the sink updates both its waypoint routing table and forwarder routing table as described above. Now, the sink is ready to accept data packets.

Reliable transmission

This protocol provides reliable packet delivery for unicast transmission similar to other reliable transmission protocols. Data are cached in the sender until an ACK is received from the receiver. If no ACK is received within a timeout period, an error report is generated and the route update phase comes into existence.

Route update

Due to the dynamic nature of sensor nodes, links on a route may fail. Route maintenance is the mechanism used to handle route breaks. In flat reactive routing protocols, when a link failure occurs, typically a route error message is sent to the source and the old route is torn down. The source then starts another route discovery. Route maintenance in this manner creates much broadcast message overhead. On the other hand, a route in this REPU protocol is composed of segments. Thus, a broken route can be repaired locally, at the level of the broken segment or at the level of a few segments near the broken segment. We divide a route into segments by selecting a number of waypoint nodes from the route. The source and the destination are also waypoint nodes. The advantages of this approach are twofold. First, since routes are maintained as segments, a broken route can be fixed locally at the level of a segment. When a route breaks, usually only a few hops are broken and other hops are still intact. Fixing a broken route within a segment extends the lifetime of the route and saves expensive, time-consuming global route discoveries. Thus, routing overhead is reduced and the performance is improved. Second, the length (hop count) of each segment on a route can be different and lengths of segments on different routes can be different. This makes our protocol an adaptive routing scheme, which is important to the sensor networks where various network scenarios exist.

Route repair

From Fig.5, the path formed between the sink and the source is 'sink'-*A*-*B*-*F*-'source' and among them nodes 'sink', *B* and 'source' are waypoint nodes. When there is a link failure between nodes *B* and *F* as shown in Fig.6, the routing path is broken, and a new path has to be set up.

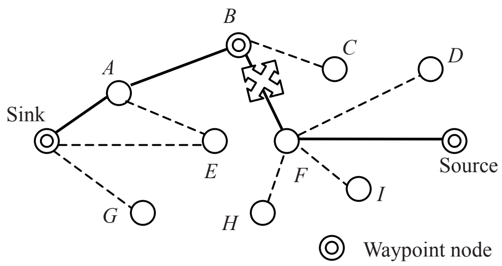

Fig.6 A path failure as represented by the cross in the solid line

The path setup phase does not need to be initiated from the sink node in the REPU protocol; since *B* is the waypoint node, it initiates the path setup phase and finds a node between nodes *B* and 'source'. The updated path is shown in Fig.7. Only the forwarding nodes are changed and the corresponding routing tables are updated. If there is a fault in the waypoint node, i.e., when a forwarding node finds the next hop node (waypoint) unreachable, it sends a route error message to its precursor node, which is a waypoint node, and initiates the path setup phase; the routing tables are updated accordingly.

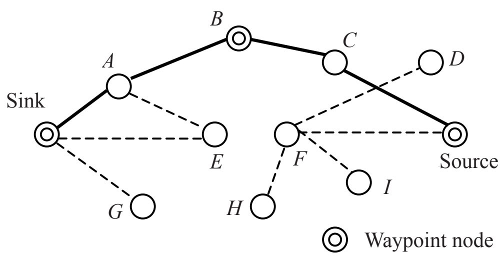

Fig.7 Route update. The solid line shows the modified path after successful route repair

PERFORMANCE ANALYSIS

We simulated energy-efficient routing with novel path update on GloMoSim (Takai *et al.*, 1999), a scalable discrete-event simulator developed by UCLA. This software provides a high-fidelity simulation for wireless communications with detailed propagation, radio and MAC layers. We compared energy-efficient routing with two popular sensor network routing protocols: SPEED (He *et al.*, 2003), a QoS routing protocol for sensor networks that provides soft real-time end-to-end guarantees, and AODV (Perkins and Royer, 1999), a routing scheme that forwards data along a single path via route request and route reply messages.

Simulation model

The GloMoSim library (Takai *et al.*, 1999) was used for protocol development in sensor networks. The library is a scalable simulation environment for wireless network systems using the parallel discrete event simulation language PARSEC. The distributed coordination function (DCF) of IEEE 802.11 was used as the MAC layer in our experiments. It uses request-to-send (RTS) and clear-to-send (CTS) control packets to provide virtual carrier sensing for unicast data packets to overcome the well-known hidden terminal problem. Each data transmission is followed by ACK. Around 100 sensor nodes were uniformly distributed over a 5 km×5 km area. All the sensor nodes were assumed to have an initial energy of 10 J and then we injected the network with 1000 randomly generated packets. The process was also performed for various energy levels. The simulation parameters are shown in Table 1. The log-distance path loss model was used and the path loss exponent was set to 4.0.

Table 1 Application parameter configuration

Parameter	Value
Transmission range	250 m
Simulation time	1000 s
Topology size	5 km×5 km
Number of sensors	100
Number of sinks	1
Traffic type	Constant bit rate
Packet rate	5 packets/s
Packet size	512 bytes
MAC layer	IEEE 802.11
Bandwidth	2 Mbps
Transmit power dissipation	660 mW
Receive power dissipation	395 mW
Idle power	35 mW
Node placement	Random
Initial energy in batteries	10 J
Signal strength threshold	−80 dBm
Energy threshold	0.001 mJ
SEGMENT_LENGTH	2

When a packet arrives, the algorithm will be invoked to compute the paths. If the algorithm cannot return a solution or the energy level of the nodes cannot satisfy the requirement imposed by the packet size, this packet will be rejected. The simulation is run for 1000 s, therefore each protocol has enough time to discover the route from the sink to the source and to produce a substantial amount of data traffic. For the evaluation of protocols, three metrics have been chosen, i.e., average energy consumption, data delivery ratio, and average delay. Each of the above metrics is evaluated as a function of the topology size, the number of nodes deployed, and the data load of the network.

Energy calculation

The power consumption model of the radio in embedded devices must take both transceiver and start-up power consumption into account along with an accurate model of the amplifier. The latter actually becomes dominant with small packet sizes and long transition time to the receiver power consumption because of frequency synthesizer settle-down time. The model for radio power consumption for energy per bit (e_{b}) is defined as

$$e_{\mathrm{b}} = e_{\mathrm{TE}} + e_{\mathrm{RE}} + E_{\mathrm{dec}} / l, \tag{1}$$

where e_{TE} and e_{RE} are the transmitter and receiver power consumptions per bit, respectively, E_{dec} is the energy required for decoding a packet, and l is the payload length in bits. The encoding of data is assumed to be negligible. This model takes into account the energy needed to transmit a frame from a transmitter to a receiver over a single hop. In this paper we extend the model to multihop scenarios and with different traffic models. The term e_{TE} from Eq.(1) with optimal power control can be represented as

$$e_{\mathrm{TE}} = e_{\mathrm{TC}} + e_{\mathrm{TA}} d^{\beta}, \tag{2}$$

where e_{TC} is the power consumption of the transmitter electronics, e_{TA} is the power consumption of the transmit amplifier, d is the transmission distance, and β is the path loss exponent. The energy used by the transmitter is further divided into

$$e_{\mathrm{TA}} = \frac{SNR_i \cdot NF_{\mathrm{RS}} \cdot N_{\mathrm{o}} \cdot BW \cdot (4\pi/\lambda)^{\beta}}{G_{\mathrm{ant}} \cdot \eta_{\mathrm{amp}} \cdot R_{\mathrm{bit}}}, \tag{3}$$

where i represents the ith sensor node, SNR_i is the signal-to-noise ratio at the receiver, NF_{RS} is the receiver noise, N_{o} is the thermal noise floor in 1 Hz bandwidth, BW is the channel noise bandwidth, λ is the wavelength, G_{ant} is the antenna gain, η_{amp} is the transmitter amplifier efficiency, and R_{bit} is the raw bit rate. This expression for e_{TA} can be used for those cases where a particular hardware configuration is considered.

Along with the above equations we made assumptions about the radio parameters used in the analysis of the paper as shown in Table 2.

Table 2 Parameters used in energy calculation

Parameter	Value	Parameter	Value
β	2	SNR_i	11 dB
G_{ant}	−20 dB	NF_{RS}	10 dB
η_{amp}	0.2	N_{o}	4.17e-21 J
R_{bit}	2.50e+5 bps	λ	0.125 m
e_{TE}	1.45e-8 J	BW	2 Mbps

Performance metrics

In the simulations data were collected for the three metrics, namely average energy consumption, data delivery ratio and average delay.

Average energy consumption (E_{a}): The average energy consumption is calculated across the entire topology. It measures the average difference between the initial level of energy and the final level of energy that is left in each node. Let E_{i} be the initial energy level of a node, E_{f} the final energy level of a node and N the number of nodes in the simulation. Then

$$E_{\mathrm{a}} = \frac{1}{N}\sum_{i=1}^{N}\left(E_{\mathrm{i}\,i} - E_{\mathrm{f}\,i}\right). \tag{4}$$

This metric is important because the energy level that a network uses is proportional to the network lifetime. The lower the energy consumption, the longer the network lifespan.

Data delivery ratio (R): This metric represents the ratio of the number of data packets sent by the source to the number of data packets received by the sink.

$$R = \frac{\text{Successfully delivered data}}{\text{Required data}}. \tag{5}$$

This metric indicates both the loss ratio of the routing protocol and the effort required to receive data. In the ideal scenario the ratio should be equal to 1. If the ratio falls significantly below the ideal ratio, it could be an indicator of some faults in the protocol design. However, if the ratio is higher than the ideal ratio, it is an indicator that the sink receives a data packet more than once. It is not desirable because reception of duplicate packets consumes the network's valuable resources. The relative number of duplicates received by the sink is also important because based on that number the sink can possibly take an appropriate action to reduce the redundancy.

Average delay (D): It is defined as the average time between the moment a data packet is sent by a data source (t_{source}) and the moment the sink receives the data packet (t_{sink}).

$$D=\frac{1}{N}\sum_{i=1}^{N}(t_{\text{sink }i}-t_{\text{source }i}). \quad (6)$$

This metric defines the freshness of data packets.

Simulation results

The delivery ratios of all the three routing protocols increase as the node density increases. When the node density is high, there are more nodes available for data forwarding, and this increases the delivery ratio. Fig.8 shows that AODV offers the least packet delivery ratios, followed by SPEED; it did not adapt well to an increase in network size. The use of REPU maintained constant delivery ratios throughout the simulated scenarios. For networks with 100 nodes, AODV delivers 40% data packets, SPEED delivers around 70%, but REPU protocol shows a high packet delivery ratio of 95% due to the impact of the process it uses to create a routing path.

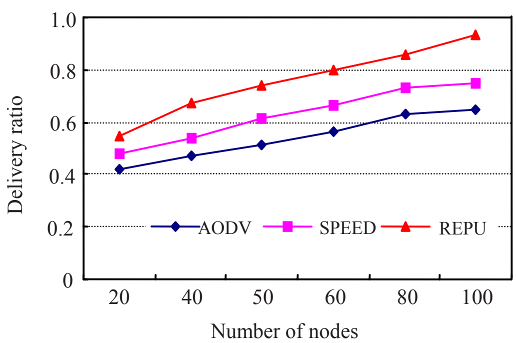

Fig.8 Delivery ratio vs. node density

Under energy constraints, it is vital for sensor nodes to minimize energy consumption in radio communication to extend the lifetime of sensor networks. From the results shown in Fig.8, we infer that energy-efficient routing tends to reduce the number of hops in the route, thus reducing the energy consumed for transmission. AODV performs the worst as a consequence of sending out many control packets. SPEED only takes delay into account, which leads to longer routes. Fig.9 shows that the SPEED protocol has nearly the same energy consumption as REPU when the node density is less than 20. When the node density is increased further, SPEED consumes more energy than REPU. At the same time, the nodes in the REPU network have consumed almost a fixed amount of energy, since only a limited number of nodes are used to route information while the rest are in sleep state. For a network with 100 nodes, the total energy consumption percentage is about 33% for the energy-aware routing, 42% for SPEED, and 58% for AODV. Therefore, we conclude that REPU will consume less energy than other protocols.

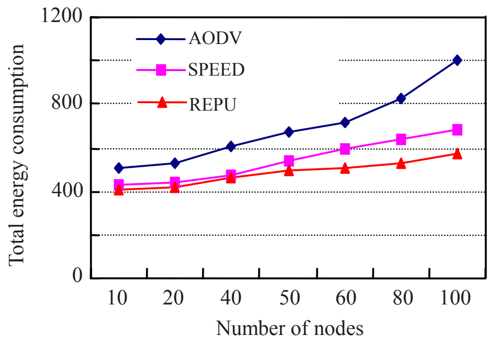

Fig.9 Energy consumption vs. node density

Fig.10 shows the performance of the protocols with various initial energy levels. It is found that the REPU protocol confides to more rounds of communication than the SPEED and traditional flooding where the nodes close to the sink die early as the shortest path is selected.

We also studied the end-to-end delay performance of these routing protocols. The average packet delays under the three schemes are plotted in Fig.11. The maximum additional delay is approximately 2.3 s for the 250-m transmission range. This additional delay grows slowly with the increase of node population. Overall, these results have demonstrated REPU protocol's ability to sustain application performance even for large node densities.

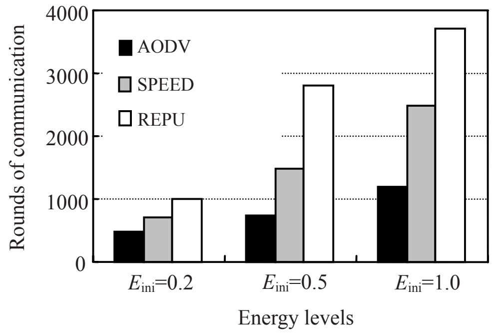

Fig.10 Performance result of protocols with various initial node energy (E_{ini})

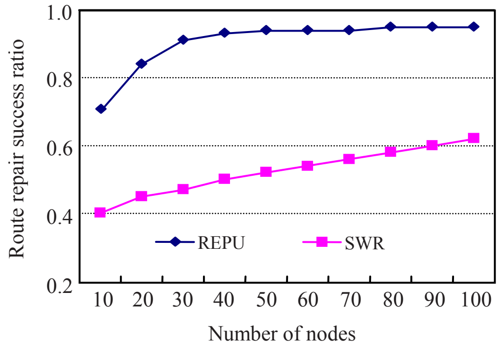

Fig.12 Route repair success ratio vs. node density

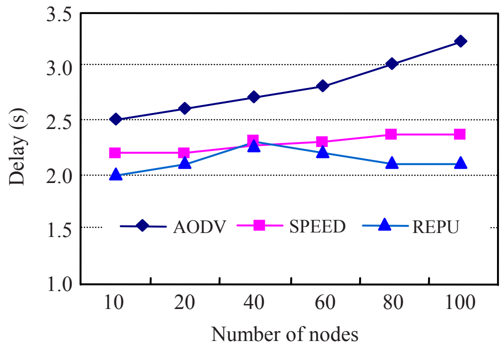

Fig.11 Average data packet delivery delay vs. node density

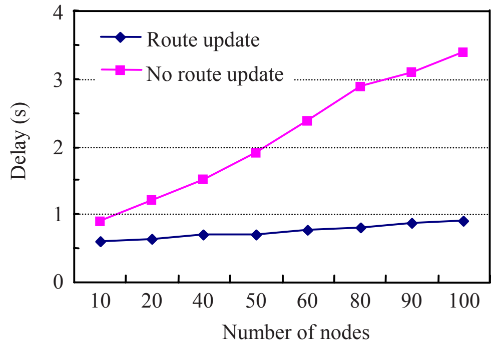

Fig.13 Delay in path setup phase vs. node density

Many other attempts at energy savings show that packet delivery performance usually decreases as a result of increased energy savings. Our results show that REPU can reduce the energy expense of communication with minimum tradeoffs in QoS.

Fig.12 shows the success ratio of route repairs in our protocol. It is observed that the success ratio increases when the node density increases. More than 80% of route repairs succeed for almost all network sizes. The results are compared with the SWR protocol (Tian and Georganas, 2003). The results also verify our discussion that when a route breaks, usually only a few hops are broken and other hops are still intact. By repairing the broken route locally, we prolong the lifetime of the route and reduce the number of global route discoveries.

A comparison of the path setup delay is plotted in Fig.13. The delay incurring in path setup phase when there occurs a failure, without and with route update mechanism for our protocol, is studied. It can be inferred from Fig.13 that the delay in identifying the alternative path increases as the node density increases when no route update mechanism is included, and that when the route update mechanism is included, the delay remains almost constant.

CONCLUSION

In WSNs, energy use is in many cases the most important constraint since it corresponds directly to the operational lifetime. We proposed a new routing model, energy-aware routing with novel route update (REPU), which maintains active routes hierarchically for WSNs. The protocol is characterized by two features. First, it offers much longer route lifetime than existing protocols. Second, it offers less routing cost than the existing protocols due to the path update mechanism. REPU divides an active route into segments by selecting a number of nodes on the route as waypoint nodes, instead of discarding the whole route and discovering a new route from the source to the destination. Only the two waypoint nodes of the broken segment need to find a new segment. This has a real performance advantage on routing overhead, energy consumption and end-to-end delay. Simulation results show that REPU uses less energy than traditional algorithms for realistic cases.

The future work we would like to focus on is how to guarantee the delivery of packets under the situation where non-uniform transmission ranges exist, and how to improve our protocol to reduce the delay. An optimal solution to this problem, especially for mobile sensor networks, is still an open question.

References

Akkaya, K., Younis, M., 2005. A survey of routing protocols in wireless sensor networks. *Elsev. Ad Hoc Network J.*, **3**(3):325-349.

Akyildiz, I., Su, W., Sankarasubramaniam, Y., Cayirci, E., 2002. A survey on sensor networks. *IEEE Commun. Mag.*, **40**(8):102-114. [doi:10.1109/MCOM.2002.1024422]

Al-Karaki, J.N., Kamal, A.E., 2004. Routing techniques in wireless sensor networks: a survey. *IEEE Pers. Commun.*, **11**(6):6-28. [doi:10.1109/MWC.2004.1368893]

Amato, G., Chessa, S., Conforti, F., Macerata, A., Marchesi, C., 2005. Health Care Monitoring of Mobile Patients. ERCIM News. Available at http://www.ercim.org/publication/Ercim_News/enw60/amato.html

Bahl, P., Padmanabhan, V.N., 2000. RADAR: An In-building RF-based User Location and Tracking System. Proc. IEEE INFOCOM, Tel Aviv, Israel, p.775-784.

Cardell-Oliver, R., Smettem, K., Kranz, M., Mayer, K., 2004. Field Testing a Wireless Sensor Network for Reactive Environmental Monitoring. Proc. Int. Conf. on Intelligent Sensors, Sensor Networks and Information Processing, Melbourne, Australia p.7-12. [doi:10.1109/ISSNIP.2004.1417429]

Cerpa, A., Elson, J., Estrin, D., Girod, L., Hamilton, M., Zhao, J., 2001. Habitat Monitoring: Application Driver for Wireless Communications Technology. Proc. ACM SIGCOMM Workshop on Data Communication in Latin America and the Caribbean, Costa Rica, p.20-41.

Deb, B., Bhatnagar, S., Nath, B., 2003. ReInForM: Reliable Information Forwarding Using Multiple Paths in Sensor Networks. Proc. 28th Annual IEEE Int. Conf. on Local Computer Networks, Bonn/Königswinter, Germany, p.406-415. [doi:10.1109/LCN.2003.1243166]

Delin, K.A., Harvey, R.P., Chabot, N.A., Jackson, S.P., Adams, M., Johnson, D.W., Britton, J.T., 2003. Sensor Web in Antarctica: Developing an Intelligent, Autonomous Platform for Locating Biological Flourishes in Cryogenic Environments. Proc. 34th Lunar and Planetary Science Conf., League City, Texas, p.1929-1930.

Demirbas, M., Chow, K.Y., Wan, C.S., 2006. INSIGHT: Internet-sensor Integration for Habitat Monitoring. Proc. Int. Symp. on a World of Wireless, Mobile and Multimedia Networks, Taormina, Italy, p.20-29.

Dulaman, S., Nieberg, T., Wu, J., Havinga, P., 2003. Trade-off between traffic overhead and reliability in multipath routing for wireless sensor networks. *IEEE Wirel. Commun. Network.*, **3**:1918-1922.

Essa, I.A., 2000. Ubiquitous sensing for smart and aware environments. *IEEE Pers. Commun.*, **7**(5):47-49. [doi:10.1109/98.878538]

Gsottberger, Y., Shi, X., Stromberg, G., Sturm, T.F., Weber, W., 2004. Embedding Low-cost Wireless Sensors into Universal Plug and Play Environments. Proc. 1st European Workshop on Wireless Sensor Networks, Berlin, Germany, p.291-306.

He, T., Stankovic, J.A., Lu, C.Y., Abdelzaher, T., 2003. SPEED: A Stateless Protocol for Real-time Communication in Sensor Networks. Proc. Int. Conf. on Distributed Computing Systems, Providence, RI, USA, p.46-55.

Heinzelman, W., Kulik, J., Balakrishnan, H., 1999. Adaptive Protocols for Information Dissemination in Wireless Sensor Networks. Proc. 5th Annual ACM/IEEE Int. Conf. on Mobile Computing and Networking, Seattle, WA, p.174-185. [doi:10.1145/313451.313529]

Heinzelman, W., Chandrakasan, A., Balakrishnan, H., 2000. Energy Efficient Communication Protocol for Wireless Sensor Networks. Hawaii Int. Conf. System Sciences, Maui, Hawaii, p.3005-3015.

Intanagonwiwat, C., Govindan, R., Estrin, D., 2000. Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks. Proc. 6th Annual Int. Conf. on Mobile Computing and Networking, Boston, p.56-66. [doi:10.1145/345910.345920]

Jensen, D., 2002. SIVAM: communication, navigation and surveillance for the Amazon. *Avion. Mag.* [Online] Available at http://www.aviationtoday.com/av/categories/military/12730.html

Lindsey, S., Raghavendra, C.S., 2002. PEGASIS: Power Efficient Gathering in Sensor Information Systems. Proc. IEEE Aerospace Conf., Big Sky, Montana, **3**:1125-1130. [doi:10.1109/AERO.2002.1035242]

Mainwaring, A., Polastre, J., Szewczyk, R., Culler, D., Anderson, J., 2002. Wireless Sensor Network for Habitat Monitoring. Proc. 1st ACM Int. Workshop on Wireless Sensor Networks and Applications, Atlanta, p.88-97. [doi:10.1145/570738.570751]

Manjeshwar, A., Agrawal, D.P., 2001. TEEN: A Protocol for Enhanced Efficiency in Wireless Sensor Networks. Proc. 1st Int. Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, San Francisco, California, p.301-309.

Martinez, K., Hart, J.K., Ong, R., 2004. Sensor network applications—environmental sensor networks. *Computer*, **37**(8):50-56. [doi:10.1109/MC.2004.91]

Mukhopadhyay, S., Panigrahi, D., Dey, S., 2004. Model Based Error Correction for Wireless Sensor Networks. Proc. 1st Annual IEEE Communications Society Conf. on Sensor and Ad Hoc Communications and Networks, Santa Clara, California, p.575-584.

Neha, J., Madathil, D.K., Agrawal, D.P., 2003. Exploiting Multi-path Routing to Achieve Service Differentiation in Sensor Networks. Proc. 11th IEEE Int. Conf. on Networks, Atlanta, Georgia, p.681-686.

Perkins, C.E., Royer, E.M., 1999. Ad-hoc on Demand Distance Vector Routing. Proc. 2nd IEEE Workshop on Mobile Computing Systems and Applications, New Orleans, Lousiana, p.90-100.

Schwiebert, L., Gupta, S.K.S., Weinmann, J., 2001. Research Challenges in Wireless Networks of Biomedical Sensors. Proc. 7th Annual Int. Conf. on Mobile Computing and

Networking, Rome, Italy, p.151-165.
Shah, R.C., Rabaey, J.M., 2002. Energy Aware Routing for Low Energy Ad Hoc Sensor Networks. Proc. IEEE Wireless Communications and Networking Conf., Orlando, FL, **1**:17-21.
Sohrabi, K., Gao, J., Ailawadhi, V., Pottie, G.J., 2000. Protocols for self-organization of a wireless sensor network. *IEEE Pers. Commun.*, **7**(5):16-27. [doi:10.1109/98.878532]
Son, B., Her, Y.S., Kim, J.G., 2006. A design and implementation of forest-fires surveillance system based on wireless sensor networks for South Korea mountains. *Int. J. Comput. Sci. Network Secur.*, **6**(9):124-130.
Srivastava, M., Muntz, R., Potkonjak, M., 2001. Smart Kindergarten: Sensor-based Wireless Networks for Smart Developmental Problem-solving Environments. Proc. ACM SIGMOBILE, 7th Annual Int. Conf. on Mobile Computing and Networking, Rome, Italy, p.132-138.
Stann, F., Heidemann, J., 2003. RMST: Reliable Data Transport in Sensor Networks. Proc. 1st IEEE Int. Workshop Sensor Network Protocols and Applications, Anchorage, Alaska, p.102-112. [doi:10.1109/SNPA.2003.1203361]
Takai, M., Bajaj, L., Ahuja, R., Bagrodia, R., Gerla, M., 1999. GloMoSim: A Scalable Network Simulation Environment. Technical Report 990027. Computer Science Department, UCLA.
Tanner, N.A., Wait, J.R., Farrar, C.R., Sohn, H., 2003. Structural health monitoring using modular wireless sensors. *J. Intell. Mater. Syst. Struct.*, **14**(1):43-56. [doi:10.1177/1045389X03014001005]
Tian, D., Georganas, N.D., 2003. Energy Efficient Routing with Guaranteed Delivery in Wireless Sensor Networks. Proc. IEEE Conf. on Wireless Communications and Networking, Phoenix, Arizona, **3**:1923-1929.
Tilak, S., Abu-Ghazaleh, N.B., Heinzelman, W., 2002. A taxonomy of wireless micro-sensor network models. *Mob. Comput. Commun. Rev.*, **6**(2):28-36. [doi:10.1145/565702.565708]
Ulmer, C., Yalamanchili, S., Alkalai, L., 2003. Wireless Distributed Sensor Networks for In-situ Exploration of Mars. Work in Progress for NASA Technical Report. Georgia Institute of Technology.
Wan, C.Y., Campbell, A.T., 2005. PSFQ: a reliable transport protocol for wireless sensor networks. *IEEE J. Sel. Areas Commun.*, **23**(4):862-872. [doi:10.1109/JSAC.2005.843554]
Xu, N., Rangwala, S., Chintalapudi, K.K., Ganesan, D., Broad, A., Govindan, R., Estrin, D., 2004. A Wireless Sensor Network for Structural Monitoring. Proc. 2nd Int. Conf. on Embedded Networked Sensor Systems, Baltimore, p.13-24. [doi:10.1145/1031495.1031498]
Xu, Y., Heidemann, J., Estrin, D., 2001. Geography-informed Energy Conservation for Ad Hoc Routing. Proc. Int. Conf. on Mobile Computing and Networking, Rome, Italy, p.70-84.
Ying, C., Lu, Q., Liu, Y., Shi, M., 2000. Routing Protocols Overview and Design Issues for Self-organized Network. Proc. Int. Conf. on Communication Technology, Beijing, China, **2**:1298-1303.
Yu, Y., Estrin, D., Govindan, R., 2001. Geographical and Energy-aware Routing: A Recursive Data Dissemination Protocol for Wireless Sensor Networks. Technical Report. Computer Science Department, UCLA.
Zhang, H., Arora, A., Choi, Y., Gouda, M.G., 2005. Reliable Bursty Convergecast in Wireless Sensor Networks. Proc. 6th ACM Int. Symp. on Mobile Ad Hoc Networking and Computing, Urbana, Champaign, USA, p.266-276. [doi:10.1145/1062689.1062724]