INTERNATIONAL JOURNAL OF COMMUNICATION SYSTEMS
Int. J. Commun. Syst. 2007; **20**:747–766
Published online 14 September 2006 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/dac.843

An energy-efficient multipath routing protocol for wireless sensor networks

Ye Ming Lu[‡] and Vincent W. S. Wong*,[†]

Department of Electrical and Computer Engineering, The University of British Columbia, 2332 Main Mall, Vancouver, Canada V6T 1Z4

SUMMARY

The energy consumption is a key design criterion for the routing protocols in wireless sensor networks. Some of the conventional single path routing schemes may not be optimal to maximize the network lifetime and connectivity. In this paper, we propose a distributed, scalable and localized multipath search protocol to discover multiple node-disjoint paths between the sink and source nodes. We also propose a load balancing algorithm to distribute the traffic over the multiple paths discovered. We compare our proposed scheme with the directed diffusion, directed transmission, *N*-to-1 multipath routing, and the energy-aware routing protocols. Simulation results show that our proposed scheme has a higher node energy efficiency, lower average delay and control overhead than those protocols. Copyright © 2006 John Wiley & Sons, Ltd.

Received 13 January 2006; Revised 6 June 2006; Accepted 30 June 2006

KEY WORDS: multipath routing; wireless sensor networks; load balancing

1. INTRODUCTION

Wireless sensor networks (WSNs) consist of densely deployed sensor nodes, which have limited computational capabilities, power supply, and communication bandwidth. The potential applications of sensor networks widely span both civil and military domains. For military applications, WSNs can be used for surveillance in battlefields. For civil applications, the sensor networks can be used to monitor light, temperature, humidity and other environmental factors

---

*Correspondence to: Vincent W. S. Wong, Department of Electrical and Computer Engineering, The University of British Columbia, 2332 Main Mall, Vancouver, Canada V6T 1Z4.
[†] E-mail: vincentw@ece.ubc.ca
[‡] E-mail: ymlu@ece.ubc.ca

Contract/grant sponsor: Natural Sciences and Engineering Research Council of Canada; contract/grant number: 261604-03

Copyright © 2006 John Wiley & Sons, Ltd.

that affect the habitat of endangered species. Other applications of WSNs can be found in Reference [1].

Depending on the network structure adopted, the routing protocols for WSNs can be classified into *flat network routing*, *hierarchical network routing*, and *location-based routing* [2]. In *flat network routing*, all nodes have equal functionality and they co-operate to perform the sensing tasks. The *sensor protocols for information via negotiation* (SPIN) [3, 4] and directed diffusion [5] fall into this category. The *hierarchical network routing* divides the network into clusters or grids in order to achieve scalability and energy efficiency. The *low-energy adaptive clustering hierarchy* (LEACH) [6] is an example of hierarchical network routing protocol. The *location-based routing* relies on the node positions, which can be obtained from *global positioning system* (GPS) device attached to the sensor to handle the data routing. The *geographic adaptive routing* (GAF) [7] and *geographic and energy-aware routing* (GEAR) [8] are two examples of the location-based routing protocol.

Given the adopted network structure, the routing protocols for WSNs can operate in different ways. That can be divided into *negotiation-based*, *query-based*, *QoS-based*, and *multipath-based*. The *negotiation-based* protocols have the objective to eliminate redundant data by including high-level data descriptors in the message exchanged. The sensor node can make communication decisions based on the data descriptors and the energy level of its battery. The SPIN [3, 4] protocol is an example of this type of protocol. For *query-based* protocols, such as the directed diffusion [5], the communication is initiated by the sink node that broadcasts query for data over the network. The source node sends the data back to the sink node if it has data that matches the query. The *QoS-based* protocols allow sensor nodes to balance between the energy consumption and certain pre-determined QoS metrics before they deliver the data to the sink node. The sequential assignment routing is one of the *QoS-based* protocols. Finally, the *multipath-based* routing protocols, such as the schemes proposed in References [9–11], tend to enhance the reliability through the use of multiple paths. The data transmission relies mostly on the optimal path. The alternative path is used only when the nodes on the primary route fail. Although the existing single-path approach is flexible, simple and scalable, nodes may deplete their energy supply at a faster rate. This may result in early network partition [2].

The multipath routing technique was initially used in wired networks for its reliability and its ability to balance traffic load over the network [12–14]. In recent years, such technique is extended to wireless *ad hoc* or sensor networks with objectives to achieve better energy efficiency and network robustness in case of node failures.

In [15, 16], a multipath extension of *dynamic source routing* (DSR) and *ad hoc on-demand distance vector* (AODV) were proposed to improve the energy efficiency of *ad hoc* networks by reducing the frequency of route discovery. The directed diffusion [5] is a data-centric routing scheme. The flooding of interest by sinks allows the gradients to be set up within the network. All nodes in the network maintain an interest cache, which associates the gradient with distinct interests. The hop-by-hop routing decision of low-rate exploratory data from the source is made based on the interest cache of each node. The sink, by collecting exploratory data arrived in low rate via different paths, sends reinforcement through one particular path to ask the source node to transmit data at a higher rate. The in-network aggregation combines the data from different sources for the same target in order to save energy and prolong network lifetime by eliminating the redundancy.

An *N*-to-1 multipath discovery protocol is proposed in Reference [17]. The route discovery process finds different node-disjoint paths between a sink and a source node. The search process

Copyright © 2006 John Wiley & Sons, Ltd.

DOI: 10.1002/dac

is carried in a spanning tree structure with the sink node as the root. The protocol allows the node to find alternative routes that belong to another branch. These alternative routes are used to distribute traffic in order to improve the reliability and the security of the data transmission.

In Reference [11], a multipath routing approach is proposed for the directed diffusion [5] to improve the resilience to node failures. Their work explores the possibility of finding alternate paths connecting the source and sink nodes when node failures occur. Directed transmission proposed in Reference [18] is one of the probabilistic routing techniques, which are derived from the flooding. It uses a retransmission probability function to reduce redundant copies of same event data. The hop distance to the destination and the number of hops that the data packets have traversed are used as parameters. The retransmission control mechanism avoids the intensive usage of the shortest path in a certain level. However, it cannot completely eliminate the possibility to transmit redundant copies.

The energy-aware routing is proposed in Reference [9]. It uses localized flooding of request messages to find all possible routes between the sources and sinks, as well as the energy costs associated to these paths. In the routing table of the sensor node, every neighbour is associated with a transmission probability, which is computed based on the cost of the path passing through it. The scheme maintains multiple paths but uses only one of them at a time, in order to avoid stressing a particular path and extend the network lifetime. The protocol may encounter a dead-lock situation, in which data packets are being transmitted back and forth between two neighbouring nodes, if both nodes select each other as the next hop continuously. Such a situation is costly in terms of energy consumption for the network.

In Reference [19], the multipath routing is formulated as a linear programming problem with an objective to maximize the time until the first sensor node runs out of energy. The sources are assumed to be transmitting data packets at a constant rate. In Reference [20], the multipath routing is formulated as a constrained optimization problem by using deterministic network calculus.

In this paper, we propose an energy-efficient multipath routing protocol for WSNs [21]. The contributions of this paper are as follows.

1. We propose a distributed, scalable and localized multipath search algorithm to discover multiple node-disjoint paths between the sink and source nodes.
2. We also propose a load balancing algorithm to distribute the traffic over the multiple paths discovered. The load balancing algorithm allows the sink node to allocate traffic over multiple paths found based on their cost, which depends on the energy levels and the hop distances of nodes along each path.
3. We compare our proposed scheme with the directed diffusion [5], directed transmission [18], the N-to-1 multipath routing [17], and the energy-aware routing [9] protocols. Simulation results show that our proposed scheme has a higher node energy efficiency, lower average delay and control overhead than those protocols.

The rest of the paper is organized as follows. Our proposed multipath data routing protocol is described in Section 2. The performance evaluation of our scheme as well as the comparisons with other protocols are presented in Section 3. We then discuss the possible enhancements of our multipath routing protocol in Section 4. Conclusions are given in Section 5.

Copyright © 2006 John Wiley & Sons, Ltd.

DOI: 10.1002/dac

2. MULTIPATH ROUTING ALGORITHM

In this section, we first describe the assumptions we made, the related definitions, and the system model. We then present the details of our multiple search protocol and the method to balance the traffic among multiple paths discovered [21].

2.1. Assumptions and definitions

We consider that M identical wireless sensor nodes are distributed randomly in a field. Each sensor node carries a radio transmitter, which has a fixed transmission range of $\Re$. We assume that the network is connected and dense. That is, given an arbitrary pair of nodes, data can be sent from one to another in a multi-hop manner. There exists multiple paths between a pair of nodes. We further assume that each sensor node is stationary and contains an internal battery to support its sensing and communication activities. This battery can neither be replaced nor recharged. At any time, a sensor node m, $m \in 1, 2, \ldots, M$, is able to acquire the residual energy level $e_{m,\mathrm{residual}}$ of its battery.

When a stimulus is detected (or an event occurs), the surrounding nodes first exchange the information and select one of them to be the source node. The source node has the responsibility to aggregate data from the neighbouring nodes and to transmit the aggregated data to the sink node. Various data aggregation algorithms have proposed in the literature (e.g. References [22–25]). In this paper, we use the data aggregation algorithm proposed in Reference [25]. When different events occur in different regions within the coverage area, data from different source nodes are not being aggregated along the path to the sink node.

We define a *path*, which consists of K nodes, where $K < M$, as a group of nodes that relay the data generated from the source node x to the sink node y. Since we assume that the network is dense, it is possible to have multiple routes between the source node x and the sink node y. In this case, it is possible to use multipath routing instead of single path routing. We assume that the multiple paths used are disjoint. That is, the path A, which consists of K nodes, and the path B, which consists of L nodes, are two groups mutually exclusive except for the source node x and the sink node y. We define a *link* as an abstract representation of a radio connection established between two neighbouring sensor nodes. A path A with K nodes therefore contains $(K - 1)$ links.

The link cost function is used by the node to select the next hop during the path search phase. Various link cost functions have been proposed in the literature (e.g. Reference [26]). In this paper, we use the following link cost function. Let N_a denote the neighbour set of node a, the sensor node a will choose the next hop by following the criterion:

$$\text{Next hop} = \arg\min_{b \in N_a} \{(1 - e_{b,\mathrm{residual}}/e_{b,\mathrm{init}})^{[\beta(1-(\Delta d+1)/d_{ay})]}\} \tag{1}$$

where d_{ay} is the distance in hops between node a and the sink node y; d_{by} is the distance in hops between node b and the sink node y; Δd is the difference between d_{ay} and d_{by}; $e_{b,\mathrm{init}}$ is the initial energy level of node b; $e_{b,\mathrm{residual}}$ is the residual energy level of node b; and β is the weight factor and $\beta > 1$. For most of the applications, the initial energy of the sensor nodes is the same. We include the initial energy level $e_{b,\mathrm{init}}$ so that the ratio $e_{b,\mathrm{residual}}/e_{b,\mathrm{init}}$ is less than 1.

Since Δd is the difference between d_{ay} and d_{by}, the value of Δd can only be equal to either -1, 0, or 1. That is, given the destination y, if the number of hops for the minimum hop path

Copyright © 2006 John Wiley & Sons, Ltd.

between nodes a and y is h, then the number of hops for the minimum hop path between node b (which is a neighbour of node a) and destination y is equal to either $h-1$, h, or $h+1$. Thus, $\Delta d + 1 \in \{0, 1, 2\}$.

The link cost function takes both the node energy level and the hop distance into consideration. Suppose $e_{b,\text{residual}}$ remains constant. In this case, the link cost increases when $(\Delta d + 1)$ increases. On the other hand, suppose $(\Delta d + 1)$ remains constant. In this case, the link cost increases as $e_{b,\text{residual}}$ decreases. In addition, the weight factor β adjusts the priority in the evaluation of link cost. A large β gives more weight to the node energy than to the hop distance. Figure 1 illustrates the impact on the evaluation of link cost when β varies.

The link cost is used as one of the selection criteria for the selection of next hop in our multipath protocol. In this function, we only consider the energy level of the receiver node b as it consumes energy for data reception and transmission if it is selected for forwarding. We do not take into account the energy level of node a, which is the sender in Equation (1). This is because no matter which node is selected as the next hop, node a still needs to spend the same amount of energy on data transmission.

For a path A, which consists of K nodes, the path cost p_A is the sum of individual link costs $l_{i(i+1)}$ along the path. That is,

$$p_A = l_{12} + l_{23} + \cdots + l_{K(K+1)} = \sum_{i=1}^{K-1} l_{i(i+1)} \tag{2}$$

The path cost p_A is used by our load balancing algorithm to allocate the data rate r_A through the path A.

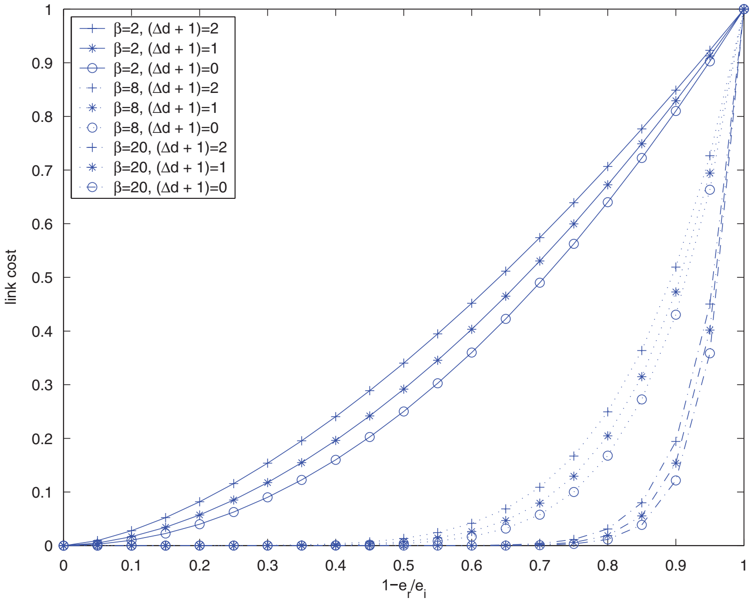


Figure 1. The evaluation of link cost with $d_{ay} = 9$.

Copyright © 2006 John Wiley & Sons, Ltd.

2.2. Multipath routing protocol

The multipath routing protocol is used to find multiple disjoint paths between a pair of sink and source nodes. It has three phases, the *initialization* phase, the *paths search* phase, and the *data transmission and maintenance* phase.

The initialization phase takes place after all sensor nodes are deployed in the target field. This phase has two objectives. First, the localized flooding of HELLO message allows all nodes to be aware of the status of their immediate neighbours. Second, the selective flooding of HELLO messages from sink nodes gives opportunities for each node to calculate its shortest distance to the sinks. Further details of the initialization phase is given in Section 2.2.1. The paths search phase follows next and it helps constructing multiple disjoint paths. We will introduce different control messages in Section 2.2.2. In Section 2.2.3, we describe how data messages are being transmitted and how the path failures are being handled by our protocol.

2.2.1. Initialization phase. The HELLO message is one of the control messages exchanged between nodes in the initialization phase. Figure 2 shows different fields within a HELLO message. The first field *message sequence* is a number generated by the message originator. The number is incremented whenever a new message is created. It is reset to 1 whenever the maximum 65 535 is reached, because the field size is 2 bytes. Combined with the node ID, it is possible to verify if the message has been received. The field *message type* carries information that it is a HELLO message. The field *sender ID* contains the node ID of the message originator. The field *node type* indicates whether the message originator is a sink, a source, or a regular sensor node. The *hop count* gives the hop distance of the message that has been passed from its originator. The *forward node ID* contains the ID of the upstream node, which forwarded the message in the previous hop. Finally, the *forward node energy level* field gives the normalized node energy level of the node that forwarded the message in the previous hop.

When the HELLO message arrives and if the message is received for the first time, each node will update its neighbouring node table with the *forward node ID* and *forward node energy level*. Next, the node verifies if the *node type* is set to be *SINK*. In such case, the *sender ID* is compared with the sink list of the node. A new entry is created in the sink table if necessary, with the hop distance updated only when it is smaller than the value recorded. Finally, the HELLO message from the sink node is re-broadcast with the fields *hop count*, *forward node ID* and *forward node energy level* updated.

The selective flooding of HELLO messages from sinks helps each node to acknowledge the existence of the sink nodes and to calculate the shortest hop distance to each sink node. At the end of the initialization phase, each node will have the sink table and the neighbouring node table updated. Each node then broadcasts a CONNECTIVITY message to its immediate

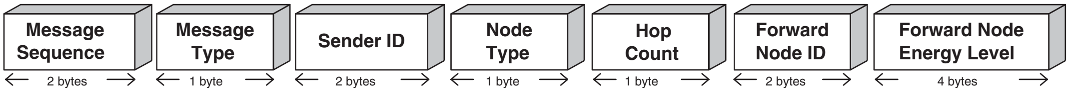


Figure 2. The format of a HELLO message.

Copyright © 2006 John Wiley & Sons, Ltd.

DOI: 10.1002/dac

neighbours. Figure 3 shows the structure of the CONNECTIVITY message. In the message, except those same fields that we have already introduced for the HELLO message, the field *sink numbers* specifies the number of sinks that the sender is aware of. The subsequent fields give in order the sink IDs and the hop distance to each of them. The receiving node will update the corresponding entry in its neighbouring node table.

2.2.2. Paths search phase. This phase is initiated when a set of nodes detect the stimulus and the selected source node begins to send the aggregated data to the sink node. Since we need to explore multiple disjoint paths, the source node unicasts one REQUEST message to every neighbouring node with a distinct route ID. We assume that the data link layer provides a reliable logical link (e.g. via the use of link layer acknowledgement). As shown in Figure 4, not all REQUEST messages will arrive to the sink node. Some of them will be dropped by the intermediate nodes in order to avoid having paths that share common nodes. Node 4 forwards the message REQ1 to node 1 rather than to node 3 as the link cost through node 1 is lower. The message REQ3 is dropped by node 1 as all its neighbours have already been selected by another path.

Figure 5 shows the format of a REQUEST message. The fields *source ID* and *sink ID* indicate the node ID of the source and sink, respectively. The *route ID* is assigned by the source node to distinguish between different routes that lead to the same sink node. The *path cost* field stores the accumulated path cost, starting from the source node. The rest of the fields carry the same information as in other control messages introduced previously.

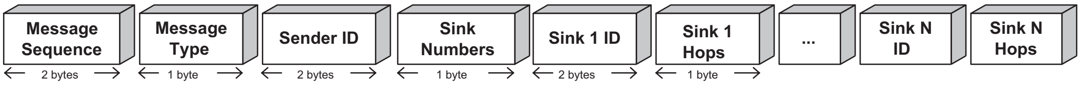


Figure 3. The format of a CONNECTIVITY message.

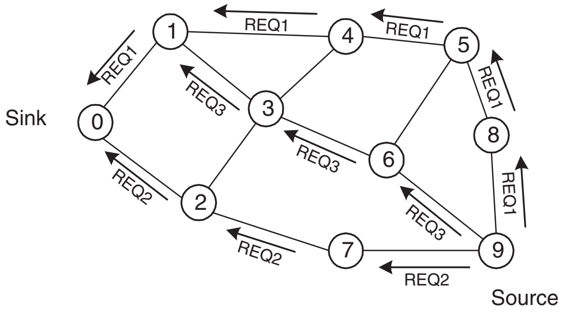


Figure 4. An example of path search.

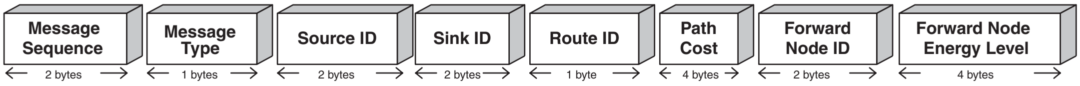


Figure 5. The format of a REQUEST message.

Copyright © 2006 John Wiley & Sons, Ltd.

DOI: 10.1002/dac

Upon reception of the REQUEST message, a regular node (i.e. an intermediate node) examines its routing table with the values in fields *source ID* and *sink ID* and creates a new entry if necessary. If the sink node indicated by *sink ID* is in the neighbouring node table, the routing table is updated and the REQUEST message is forwarded to the sink node directly with fields *forward ID* and *forward node energy level* updated. Otherwise, the node has to select one of the neighbours to forward the REQUEST message. The selection is based on two criteria. First, the neighbouring node should not have been selected for another path that connects the same pair of sink and source nodes. Second, the link cost to the selected neighbour has to be the lowest among all the available neighbours. The link cost is defined in Equation (1).

The routing table will be updated if a neighbour is selected. The table of neighbours is updated at the same time. In future path search, the node will avoid to select the neighbour that has already been used for the path that connects the same pair of sink and source nodes. Finally, the node will update the fields *path cost*, *forward node ID* and *forward node energy level* before sending the REQUEST message to the neighbour selected. If none of the neighbours satisfies the conditions, the REQUEST message will simply be dropped.

For the sink node, the received REQUEST message is processed differently. It first examines the *source ID* and creates a new entry in its source table if it is not known. It then updates the routing table with the information carried in the message. The sink node starts a request timer when it receives the first REQUEST message from a source node. The REQUEST messages arrive after the timer expires will simply be dropped. Such measure allows the path exploration to be completed within a reasonable period of time, as REQUEST messages that arrive late will include only paths with undesirable qualities (e.g. large delays and extra network resources). When the request timer expires, the sink node begins to allocate traffic to each of the path discovered. Different data rates are assigned to these paths depending on their path cost. We will present the algorithm used for rate allocation in Section 2.3. The sink node then sends the ASSIGN messages to the source node via each of the selected multipath.

Figure 6 shows the structure of the ASSIGN message. In the ASSIGN message, the field *data rate* indicates the data transmission rate assigned for the path that is specified by *route ID*. When an intermediate node receives the ASSIGN message, it searches its routing table for the entry that matches *source ID*, *sink ID* and *route ID* values. It then forwards the message to the next hop after updating the fields *forward node ID* and *forward node energy level*. The source node behaves differently when it receives the ASSIGN message. It first finds the entry specified by *sink ID* and *route ID* from its routing table. The entry is then updated with the *data rate* carried in the ASSIGN message.

2.2.3. Data transmission and paths maintenance phase. After multiple paths are discovered, the source node begins to transmit data packets with the assigned rates on each path. The DATA message carries the event data and has the fields as shown in Figure 7. The DATA message has some specific fields. The field *data count* has the value of the data counter in the source node at the time when the DATA message that related to a stimulus detected is generated. The data counter increments continuously and it resets to 0 when the maximum is reached or the event

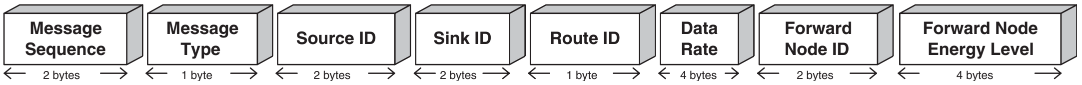


Figure 6. The format of an ASSIGN message.

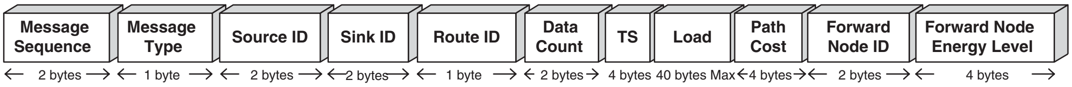


Figure 7. The format of a DATA message.

data of a new stimulus is generated. The sink node can differentiate the event data from the same source node but related to distinct stimulus with the value of the field *data count*. The field *TS* carries the timestamp, which corresponds to the time when the DATA message is created at the source node. It allows the sink node to monitor the overall packet transfer delay. Finally, the field *Load* contains the actual event data from the source node and the field *path cost* gives the accumulated path cost.

At each hop, the node can determine the next step by searching its routing table with the information carried in the DATA message, such as *source ID*, *sink ID* and *route ID*. The fields *path cost*, *forward node ID* and *forward node energy level* in the DATA message are updated before the message is being forwarded.

At the sink node, it updates the path cost in its routing table each time a DATA message arrives. The updated values help the sink node to monitor the conditions of the multiple paths being used. The initial data rate assignments for the paths may not be optimal for the duration of the connection. Usually, the path with the lowest cost is more likely to be assigned with the highest data rate initially and the nodes on that path will dissipate energy at a faster rate. Its path cost will gradually be less competitive compared with other paths. The sink node has to redistribute the data rates over paths to optimize the usage of network resources. The redistribution is triggered when the original route with the lowest cost has its path cost increased to a pre-determined threshold. The sink node will then adjust the traffic flows and notify the source node with the ASSIGN messages.

In order to detect the path failure, the sink also monitors the inter-arrival delay of data packets on each path. When the delay is above a pre-determined threshold, the sink presumes that the path is broken. If the number of current working paths is equal to or lower than two, the sink will send a RESET message to the source through the optimal path to re-initiate the paths search phase. Otherwise, the sink re-adjusts the data rate allocation over other functional routes. This mechanism can avoid having the path search phase being invoked frequently.

2.3. Load balancing algorithm

We assume that there exists N disjoint paths between a source node x and a sink node y. The requested data rate to be arrived at the sink node y via all these multipaths is R bits/sec. Let r_j be the data rate allocated to path j, we have

$$\sum_{j=1}^{N} r_j = R \quad \text{where } r_j \geqslant 0 \tag{3}$$

For a path j, the product of the path cost p_j and the data rate allocated r_j gives the path cost rate c_j. The overall system cost C to transmit data with rate R between a sink node and a source node can be expressed as:

$$C = \sum_{j=1}^{N} c_j = \sum_{j=1}^{N} r_j p_j \quad \text{where } r_j \geqslant 0 \tag{4}$$

Copyright © 2006 John Wiley & Sons, Ltd.

As we intend to improve the network energy efficiency through load balancing over multiple paths, we adopt the *Chebyshev sum inequality* in our algorithm to measure how well the transmission cost is balanced. The *Chebyshev sum inequality* is defined as follows [27]:

For two sets of distribution $\bar{a}$ and $\bar{b}$, where $\bar{a} = (a_1, a_2, \ldots, a_n), \bar{b} = (b_1, b_2, \ldots, b_n)$,
if $a_1 \geqslant a_2 \geqslant \cdots \geqslant a_n$, and $b_1 \geqslant b_2 \geqslant \cdots \geqslant b_n$,
then

$$n \sum_{k=1}^{n} a_k b_k \geqslant \left(\sum_{k=1}^{n} a_k \right) \left(\sum_{k=1}^{n} b_k \right) \tag{5}$$

We use the following *load balance ratio* Φ (also known as fairness index) to evaluate the level of load balancing over different multipaths:

$$\Phi(\bar{r}) = \frac{(\sum_{j=1}^{N} r_j p_j)^2}{N \sum_{j=1}^{N} (r_j p_j)^2} \tag{6}$$

where the vector $\bar{r}$ denotes the traffic rates allocated to all available routes and r_j is the traffic flow allocated to path j. The *load balance ratio* in Equation (6) reaches its global maximum of 1 under the condition that the traffic is perfectly balanced. This is a known property of the *Chebyshev sum inequality*.

Our traffic allocation problem can be formulated as an optimization problem:

$$\begin{aligned} &\text{Max } \Phi(\bar{r}) \\ &\text{subject to } \textstyle\sum_{j=1}^{N} r_j = R \quad \text{where } r_j \geqslant 0 \end{aligned} \tag{7}$$

To solve the above problem, we first let $k = r_j p_j$ for all j. This makes Φ to be equal to one (i.e. global maximum). By substituting $r_j = k/p_j$ into the constraint, we have

$$r_j = \frac{R}{p_j} \sum_{i=1}^{N} p_i, \quad j = 1, 2, \ldots, N \tag{8}$$

3. PERFORMANCE EVALUATION AND COMPARISON

In this section, we present the results for the performance of our proposed multipath routing protocol. We implement our multipath routing protocol in the *ns-2* network simulator and compare it with the directed diffusion [5], the energy-aware routing [9], the directed transmission [18], N-to-1 multipath routing [17], and the flooding protocols.

3.1. Simulation parameters and performance metrics

In all our simulations, we consider a square sensor field of size L. Inside the field, M static sensor nodes are deployed randomly. The value of M is varied from 50 to 250. Each node has a fixed radio range of 40 m. The node density is maintained at a constant level of $50/160^2$ nodes/m^2 ($= 1.95 \times 10^{-3}$ nodes/m^2). The positions of the source and sink nodes are shown in Figure 8. In these configurations, the sinks and sources are located far from each other. The minimum distance between any pair of sink and source is larger than $L/2$. Such settings facilitate our evaluation of the protocol where the routing path has to traverse a large area in the sensor field.

Copyright © 2006 John Wiley & Sons, Ltd.

DOI: 10.1002/dac

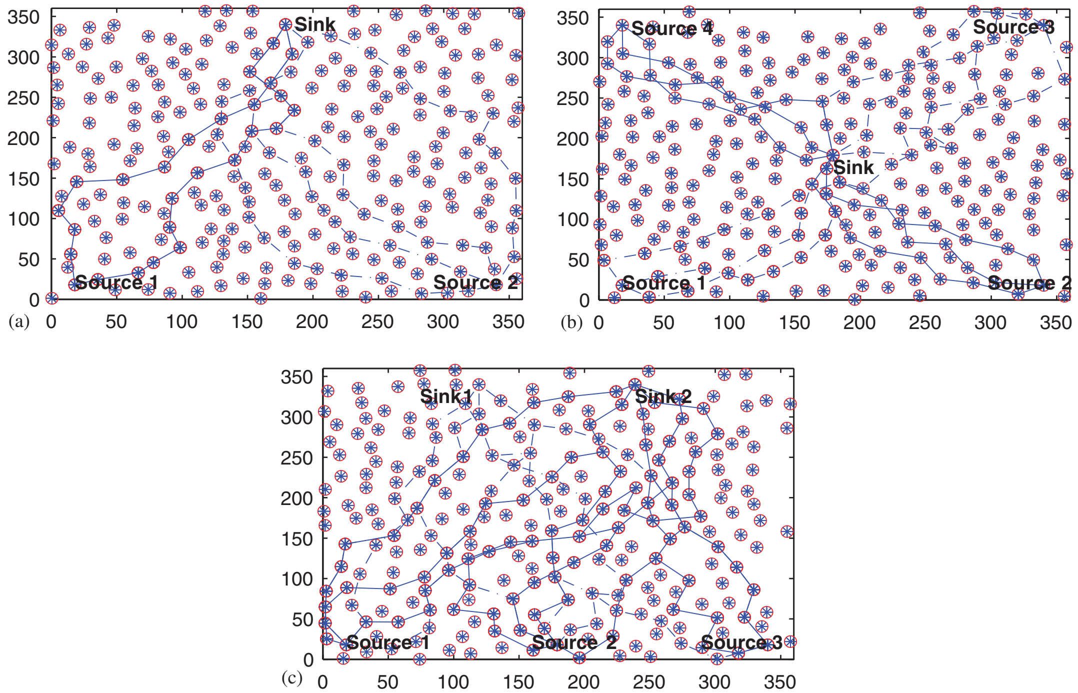


Figure 8. Configurations of sink and source nodes and examples of paths discovered with 250 nodes deployed: (a) topology setting 1: one sink and two sources; (b) topology setting 2: one sink and four sources; and (c) topology setting 3: two sinks and three sources.

We also assume that the source nodes detect different stimulus. Thus, their event data cannot be aggregated.

The data packet size is 64 bytes. We use an event-driven WSN in our experiments. After the route search phase, each source node generates data packets and sends them to the sinks through the network with a fixed rate. Unless stated otherwise, the packet data transmission rate is 1 packet per second. We use $\beta = 20$ for the link cost evaluation in Equation (1).

We adopt the *ns-2* radio energy model and assign each node with the same initial energy level of 10 J at the beginning of each simulation in order to keep the simulation time within a reasonable time period. We set the initial energy level of the sinks at 40 J as the sink usually can have its energy supply recharged or replaced in real applications. We further assume that each sensor node carries an omni antenna and the energy consumptions for idle time, transmission and reception are 35, 660 and 395 mW, respectively (the same parameters as in Reference [5]). The energy dissipation for data processing in the node is neglected in our simulations. The sinks send the HELLO messages periodically. The time interval is 30 s. We adopt the IEEE 802.11 MAC layer provided in the *ns-2* with a bandwidth of 1.6 Mbps. Every 500 ms, we obtain the log of the energy level of each node. This allows us to trace the status of energy consumption of the network. The total simulation time is 500 s. In Table I, we summarize the simulation parameters.

We use a number of metrics to evaluate the performance of our protocol. The *network lifetime* measures how long the network can sustain the data transmission from the source nodes to the

Copyright © 2006 John Wiley & Sons, Ltd.

DOI: 10.1002/dac

Table I. Simulation parameters.

Item	Value
Node density	$50/160^2$
Number of nodes	50, 100, 150, 200, 250
Data packet size	64 bytes
Control packet size	32 bytes
Idle power	35 mW
Receive power	395 mW
Transmit power	660 mW
Node initial energy	10 J
Node radio range	40 m
MAC protocol	IEEE 802.11 (CSMA/CA)
Bandwidth (802.11)	1.6 Mbps

sink nodes. There are various definitions of network lifetime proposed in the literature. Some researchers define it as the time when the first sensor node runs out of energy. Others define the network lifetime as the time when a fixed percentage of nodes run out of energy. In this paper, we define it as the time when the sink no longer receives data packets from the source. This happens when some of the intermediate relay nodes run out of energy and there are no feasible paths from the source to the sink. This time can be obtained by calculating the average interval between the first and last data packet arrivals at each sink node.

The *node energy consumption* measures the average energy dissipated by the node in order to transmit a data packet from the source to the sink. The same metric is used in the work on directed diffusion [5] to indicate the energy efficiency level of WSNs. It is calculated as follows:

$$\text{node energy consumption} = \frac{\sum_{i=1}^{M} (e_{i,\text{init}} - e_{i,\text{res}})}{M \sum_{j=1}^{S} \text{data } N_j} \tag{9}$$

where M is the number of nodes in the network, $e_{i,\text{init}}$ and $e_{i,\text{res}}$ are, respectively, the initial and residual energy levels of node i, S is the number of sink nodes and data N_j is the number of data packets received by sink j.

The *average delay* measures the average time spent to relay data packets from the source node to the sink node. The *average node energy* measures the average energy level of all nodes in the network 25 s after the data transmission has been started. It gives an indication of the network state in terms of energy consumption. Finally, we compute the *control message overhead*, which counts the average amount of control messages received and transmitted by each node in bytes. It evaluates the extra workload required to sustain the data routing for various schemes.

3.2. Network lifetime

Figure 9 shows that the network lifetime has a decreasing trend as the network size becomes large. Comparing with other schemes, the network lifetime with multipath routing has a substantial increase of 9–18% than energy-aware routing, in both data rates used with 250 nodes. We can also observe that the network lifetime with multipath routing degrades more gracefully than other routing protocols when the network size increases. It demonstrates that our routing scheme is more stable with the variation of the network size.

Copyright © 2006 John Wiley & Sons, Ltd.

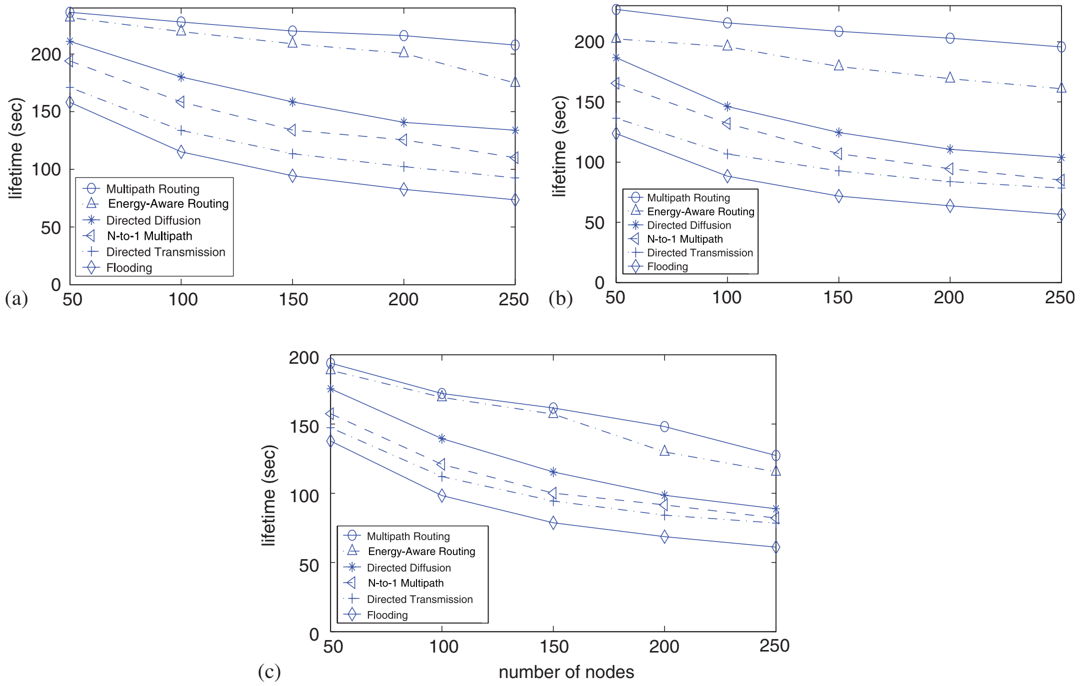


Figure 9. Average network lifetime: (a) topology setting 1: one sink and two sources; (b) topology setting 2: one sink and four sources; and (c) topology setting 3: two sinks and three sources.

The simulation results match with assumption made initially that the network lifetime can be extended by transmitting data over multiple paths simultaneously. Both energy-aware routing [9] and our multipath routing perform better than all other routing protocols we compared. The energy-aware routing uses a transmission probability mechanism to distribute the load over possible routes. Although this mechanism may not always be as efficient as our load balancing algorithm, it can reduce the traffic through a single path and avoid partition the network at an early stage. The *N*-to-1 routing [17] has smaller network lifetime than the energy-aware routing in all three topologies. The path selection in *N*-to-1 routing does not consider the node energy level during the route search. It also lacks an efficient load balancing algorithm to distribute the traffic in an energy-efficient manner. The flooding and the directed transmission [18] have the worst performance. It is due to the fact that both routing protocols do not have an efficient retransmission control mechanism. The large number of redundant data copies that are retransmitted between different sensor nodes deplete quickly the network resources.

3.3. *Average node energy consumption*

Figure 10 shows the simulation results the *node energy consumption* under different topology settings. We can observe that there is a lower node energy consumption of our multipath routing over the other schemes. The flooding is the most costly protocol; by adding a simple mechanism of retransmission probability control on top of the flooding, the directed transmission improves

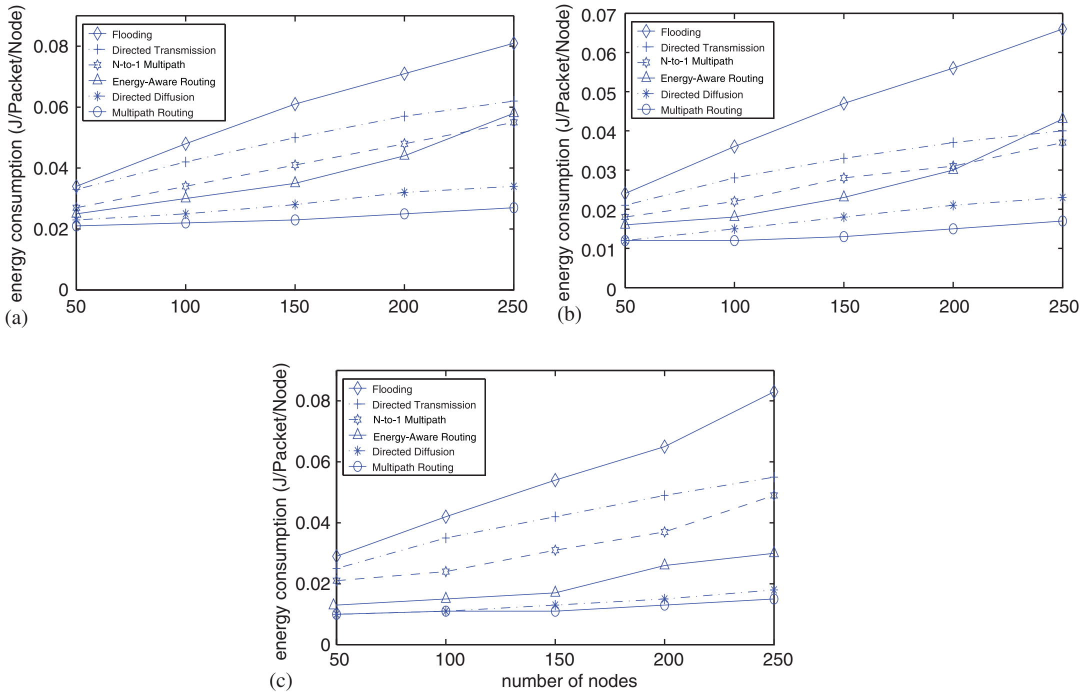


Figure 10. Average node energy consumption: (a) topology setting 1: one sink and two sources; (b) topology setting 2: one sink and four sources; and (c) topology setting 3: two sinks and three sources.

the energy efficiency. The energy-aware routing obtains further improvement by calculating the retransmission probability as function of the node energy level and the hop distance to the destination. The multipath routing and directed diffusion perform better than other protocols we examined. The better performance on directed diffusion is contributed by the capability of the protocol to find the path with the shortest delay. The periodic broadcasting of low-rate exploratory data from the source also helps changing to another route when the quality of the existing path degrades to a certain level.

Our multipath routing protocol can maintain its node energy consumption at a low level even when the network size increases. For example, in Figure 10(a), compare to directed diffusion, the improvement of multipath routing is 1–34% when the network size increases from 50 nodes to 250 nodes, with 1 sink and 2 sources. Such experimental results demonstrate that the energy efficiency of multipath routing is stable and has little impact by the increase of the network size, while the performance of other schemes degrades with larger network size.

With the same data rate, Figure 10(b) shows a better performance than Figure 10(a) for multipath routing. It is simply due to the difference of topology settings. With one sink in the centre of the field and four sources at four corners in topology 1, the average path length is significantly smaller than that in the topology 2, where one sink and two sources are in opposite edges of the field. As a result, more energy is required to deliver data to the sink in the setting of one sink and two sources.

Copyright © 2006 John Wiley & Sons, Ltd.

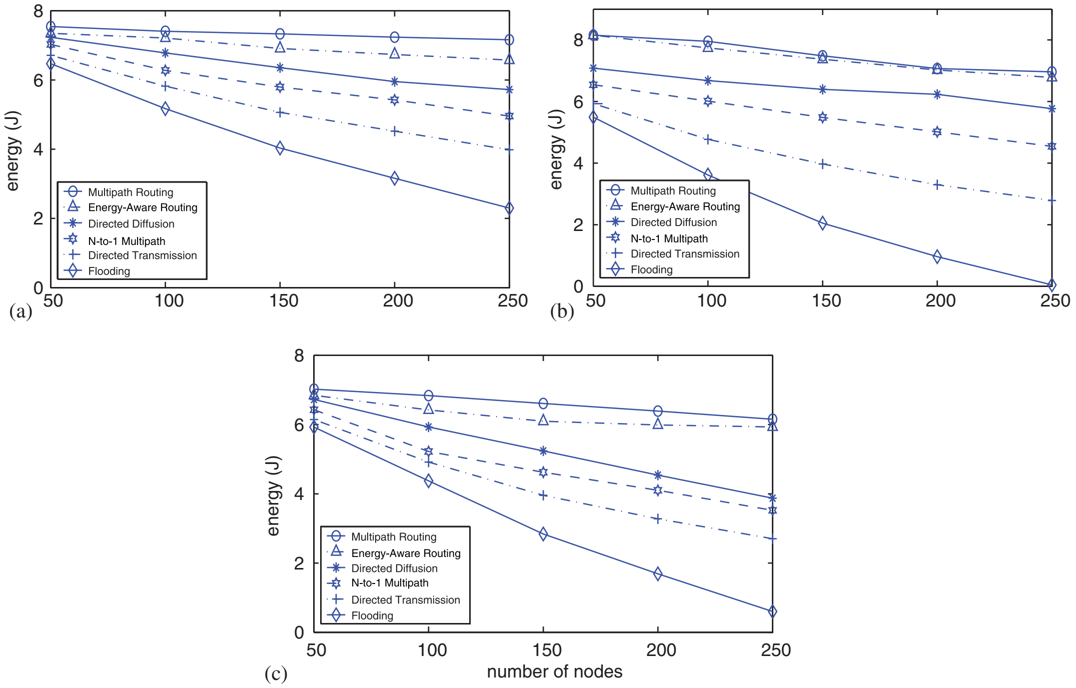


Figure 11. Average node energy level: (a) topology setting 1: one sink and two sources; (b) topology setting 2: one sink and four sources; and (c) topology setting 3: two sinks and three sources.

Figure 11 shows the *average node energy level* 25 s after the data transmission starts. We notice that the network size has an impact on the node energy level. The average node energy level decreases with larger network. It is more obvious with the flooding and the directed transmission, as they cannot completely eliminate the redundant data copies in the network. The lack of a data retransmission mechanism makes the average node energy level for flooding and directed transmission to degrade much faster than the other three protocols.

In order to study the impact of the node density on the performance of our multipath routing algorithm, we evaluate the variation of the node energy consumption as function of the node connectivity. The results are presented in Figure 12. We can observe that the average node consumption is reduced as the node connectivity increases. As the node density increases, more paths can be found between the sink and the source. This will improve the traffic load handled by each individual node, and thus reduce the average energy consumption.

3.4. Control overhead and average delay

For other routing schemes, a larger network requires more exchange of control messages to discover and construct the routes; therefore, more energy is consumed in the set-up phase. Also, a larger network implies a longer distance that separates the sink and the source nodes. More intermediate nodes are traversed before a data packet can reach the sink nodes.

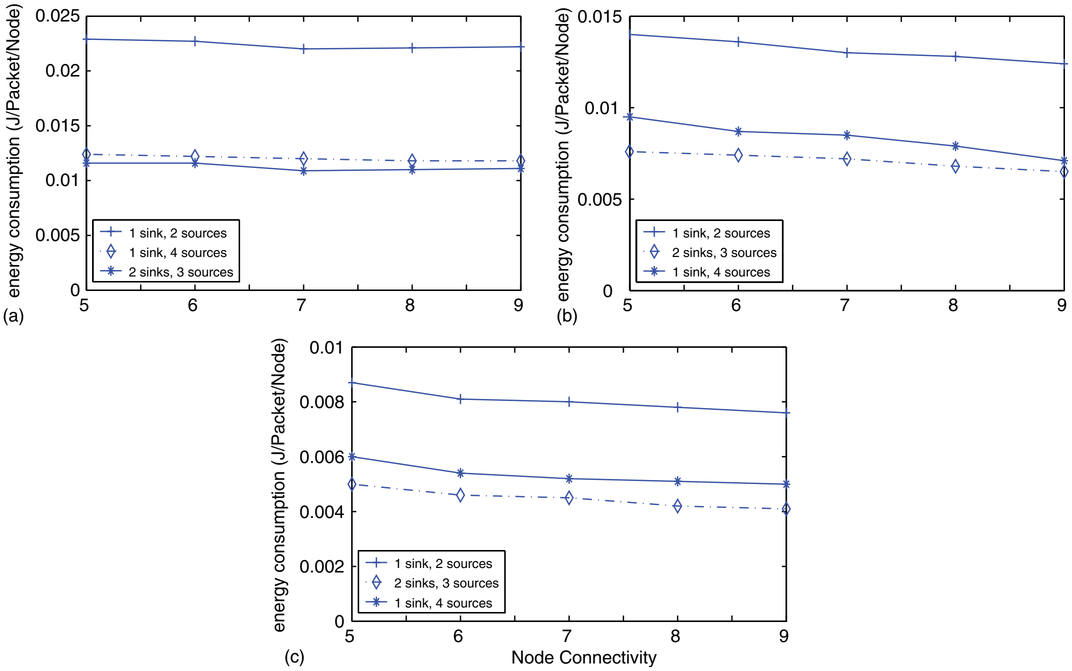


Figure 12. Average node energy consumption with 100 node in different connectivities. Data rate at (a) 1 packet/s; (b) 2 packets/s; and (c) 5 packets/s.

Figure 13 shows the *control message overhead* of different protocols. It is obtained by calculating the ratio between the average amount of control message processed by the node and the amount of data packets received by the sinks. The directed diffusion spends much more energy on transmitting and receiving control messages than any other protocols, since it requires periodic interest broadcast and path reinforcement. The multipath routing has a much lower overhead for the control message, about 70% less than the energy-aware routing [9] with the topology setting of one sink and four sources.

Figure 14 shows the results for the *average data transfer delay*. The *average delay* measures the average time spent to relay data packets from the source node to the sink node. The multipath routing has the shortest delay compared to other schemes. As we expected, data packet are routed through different node-disjoint paths with multipath routing. Hence, the network congestion can be avoided.

4. MODIFICATION AND ENHANCEMENT OF MULTIPATH ROUTING PROTOCOL

In this section, we describe the extensions for our multipath routing protocol. The first extension is to integrate data aggregation in the protocol. Currently, we assume the sensor nodes around a stimulus select one of the sensor nodes to be the source node. But the protocol does not have a

Copyright © 2006 John Wiley & Sons, Ltd.

DOI: 10.1002/dac

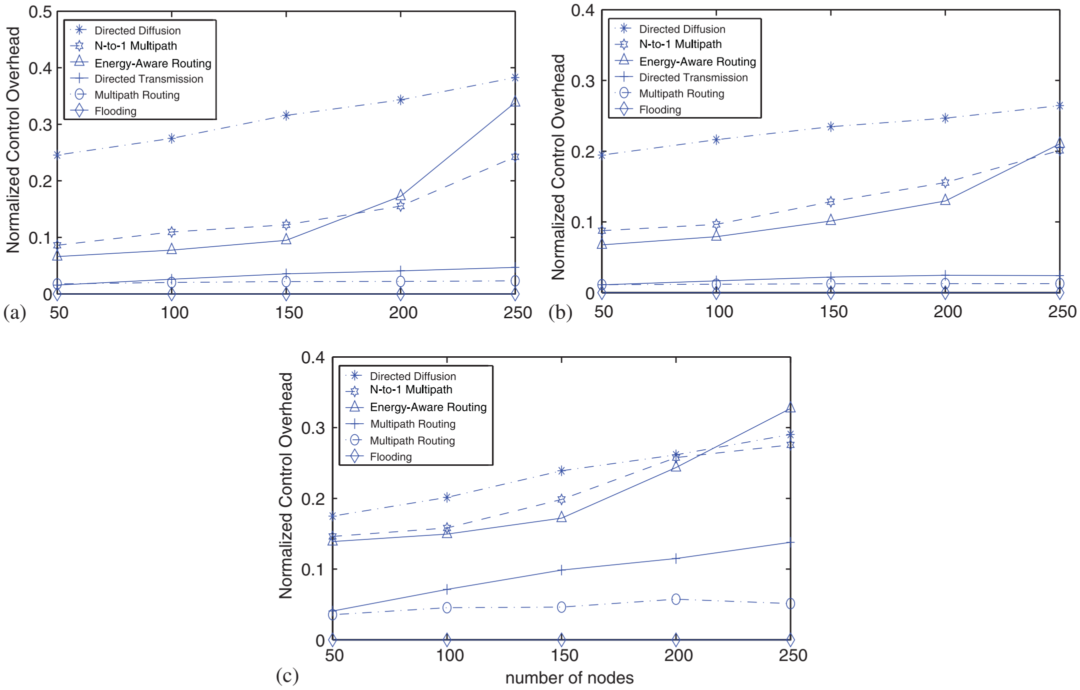


Figure 13. Ratio of control message overhead and data traffic received: (a) topology setting 1: one sink and two sources; (b) topology setting 2: one sink and four sources; and (c) topology setting 3: two sinks and three sources.

mechanism of in-network aggregation. It is possible to have two source nodes separated by an obstacle. They detect the same stimulus but are unable to aggregate their data. In such circumstance, the aggregation of event data by the intermediate nodes will save the network resources by reducing the number of redundant data in the network. The directed diffusion [5] has a mechanism of data aggregation with promising performance. Therefore, the enhancement of integrating in-network aggregation will give promising results. The key point is to make the actual control messages to be data-centric by adding descriptive information in the REQUEST and DATA messages.

Currently, our multipath routing protocol is assumed to be used for static sensor nodes. In real applications, the integration of sensor nodes with limited mobility can make the network to be more practical and robust. Therefore, the extension of our multipath routing protocol to adapt to such type of network is useful. A location update mechanism is required to allow each node to be aware of its own and its neighbours' positions constantly. It is a challenge to balance between the node energy consumption and the additional maintenance efforts that keep the node co-ordinates updated [28, 29].

Our multipath routing protocol evaluates the link cost based on the empirical measurement of the node residual energy level and its hop distance to the destination. We can improve the accuracy of the link cost evaluation and the rate allocation by interacting with the data link and physical layers, which can provide various information including the link utilization and

Copyright © 2006 John Wiley & Sons, Ltd.

DOI: 10.1002/dac

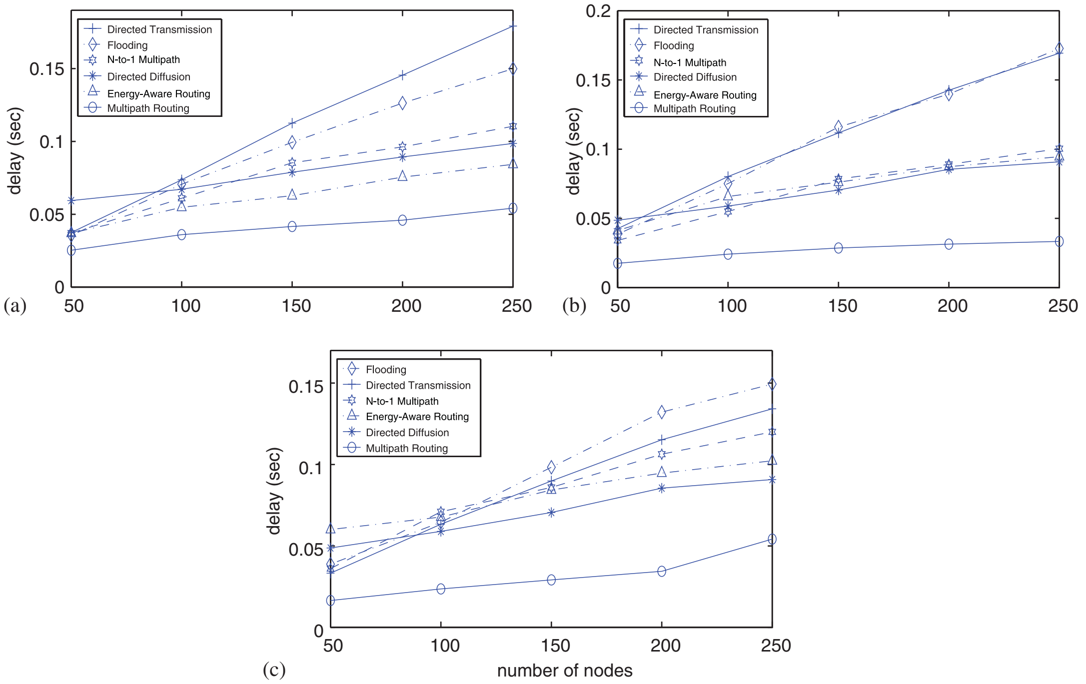


Figure 14. Average data transfer delay: (a) topology setting 1: one sink and two sources; (b) topology setting 2: one sink and four sources; and (c) topology setting 3: two sinks and three sources.

channel conditions. The cross-layer optimization may result in a better congestion control and QoS support for sensor networks [30].

5. CONCLUSIONS

In this paper, we proposed a multipath routing protocol for wireless sensor networks. The distributed multipath routing protocol is capable to search multiple node-disjoint paths. We described the format of different control messages. The proposed load balancing algorithm aims to allocate the traffic rate to each path optimally. Simulation results that our proposed scheme has a higher node energy efficiency, lower average delay and control overhead than the directed diffusion, directed transmission, *N*-to-1 multipath routing, and the energy-ware routing protocols. Further work to improve the algorithm includes the support of nodes with limited mobility.

ACKNOWLEDGEMENTS

The authors also wish to thank our colleagues, particularly Marc Lee, for his help and comments on the simulation work. This work is supported by the Natural Sciences and Engineering Research Council of Canada under grant number 261604-03.

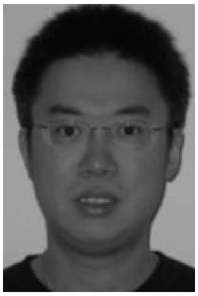
REFERENCES

1. Akyildiz I, Su W, Sankarasubramaniam Y, Cayirci E. Wireless sensor networks: a survey. *Computer Networks* 2002; **38**:393–422.
2. Al-Karaki J, Kamal A. Routing techniques in wireless sensor networks: a survey. *IEEE Wireless Communications* 2004; **11**:6–28.
3. Heinzelman W, Kulik J, Balakrishnan H. Adaptive protocols for information dissemination in wireless sensor networks. *Proceedings of ACM/IEEE MobiCom'99*, Seattle, WA, U.S.A., August 1999; 174–185.
4. Heinzelman JK ad WR, Balakrishnan H. Negotiation-based protocols for disseminating information in wireless sensor networks. *IEEE Network* 2002; **8**:169–185.
5. Intanagonwiwat C, Govindan R, Estrin D. Directed diffusion: a scalable and robust communication paradigm for sensor networks. *Proceedings of ACM MobiCom'00*, Boston, MA, U.S.A., August 2000; 56–67.
6. Heinzelman W, Chandrakasan A, Balakrishnan H. Energy-efficient communication protocol for wireless microsensor networks. *Proceedings of the 33rd International Conference on System Science* (*HICSS'00*), Hawaii, U.S.A., January 2000.
7. Xu Y, Heidemann J, Estrin D. Geography-informed energy conservation for ad-hoc routing. *Proceedings of ACM/IEEE MobiCom'01*, Rome, Italy, July 2001; 70–84.
8. Xu Y, Govindan R, Estrin D. Geographical and energy aware routing: a recursive data dissemination protocol for wireless sensor networks. *Technical Report UCLA/CSD-TR-01-0023*, UCLA Computer Science Department, May 2001.
9. Shah R, Rabaey J. Energy aware routing for low energy ad hoc sensor networks. *Proceedings of IEEE WCNC'02*, Orlando, FL, March 2002; 350–355.
10. Wu K, Harms J. On-demand multipath routing for ad hoc networks. *Proceedings of European Personal and Mobile Communications Conference* (*EPMCC*), Vienna, Austria, February 2001.
11. Ganesan D, Govindan R, Shenker S, Estrin D. Highly-resilient, energy-efficient multipath routing in wireless sensor networks. *Proceedings of ACM MobiHoc'01*, Long Beach, CA, U.S.A., October 2001.
12. Suzuki H, Tobagi F. Fast bandwidth reservation scheme with multi-link & multi-path routing in ATM networks. *Proceedings of IEEE INFOCOM'92*, Florence, Italy, May 1992.
13. Georgatsos P, Griffin D. A management system for load balancing through adaptive routing in multiservice ATM networks. *Proceedings of IEEE INFOCOM'96*, San Francisco, CA, U.S.A., March 1996.
14. Maxemchuk N. Dispersity routing in high-speed networks. *Computer Networks and ISDN System* 1993; **25**(6):645–661.
15. Nasipuri A, Das S. On-demand multipath routing for mobile ad hoc networks. *Proceedings of International Conference on Computer Communications and Networks* (*IC3N*), Boston, MA, U.S.A., October 1999.
16. Marina M, Das S. On-demand multipath distance vector routing in ad hoc networks. *Proceedings of the Ninth International Conference for Network Protocols* (*ICNP*), Riverside, CA, U.S.A., November 2001.
17. Lou W. An efficient *n*-to-1 multipath routing protocol in wireless sensor networks. *Proceedings of IEEE International Conference on Mobile Ad-hoc and Sensor Systems* (*MASS*), Washington, DC, November 2005.
18. Barrett C, Eidenbenz S, Kroc L, Marathe M, Smith J. Parametric probabilistic sensor network routing. *Proceedings of ACM WSNA'03*, San Diego, CA, September 2003; 122–131.
19. Chang J, Tassiulas L. Energy conserving routing in wireless ad-hoc networks. *Proceedings of IEEE INFOCOM'00*, Tel-Aviv, Israel, March 2000; 22–31.
20. Mao S, Panwar S, Hou Y. On optimal traffic partitioning for multipath transport. *Proceedings of IEEE INFOCOM'05*, Miami, FL, March 2005; 2325–2336.
21. Lu Y. Multipath routing algorithm for wireless sensor networks. *M.A.Sc. Thesis*, The University of British Columbia, Vancouver, Canada, December 2005 [Online]. Available: www.ece.ubc.ca/~vincentw/T/Lu.pdf
22. Ding M, Cheng X, Xue G. Aggregation tree construction in sensor networks. *Proceedings of IEEE VTC'03*, Orlando, FL, October 2003; 2168–2172.
23. Upadhyayula S, Annamalai V, Gupta SKS. A low-latency and energy-efficient algorithm for convergecast. *Proceedings of IEEE GLOBECOM'03*, December 2003; 3525–3530.
24. Zhang W, Cao G. DCTC: dynamic convoy tree-based collaboration for target tracking in sensor networks. *IEEE Transactions on Wireless Communications* 2004; **3**(5):1689–1701.
25. Lee M, Wong V. LPT for data aggregation in wireless sensor networks. *Proceedings of IEEE Globecom*, St. Louis, Missouri, December 2005.
26. Lin L, Shroff N, Srikant R. Asymptotically optimal power-aware routing for multihop wireless networks with renewable energy sources. *Proceedings of IEEE Infocom*, Miami, Florida, March 2005.
27. Hardy G, Littlewood J, Polya G. *Inequalities* (2nd edn). Cambridge University Press: Cambridge, 1988.
28. Ssu K-F, Ou C-H, Jiau H. Localization with mobile anchor points in wireless sensor networks. *IEEE Transactions on Vehicular Technology* 2005; **54**:1186–1197.
29. Hu L, Evans D. Localization for mobile sensor networks. *Proceedings of ACM MobiCom*, Philadelphia, PA, September 2004; 45–57.
30. Berry R, Yeh E. Cross-layer wireless resource allocation. *IEEE Signal Processing Magazine* 2004; **21**:59–68.

Copyright © 2006 John Wiley & Sons, Ltd.

AUTHORS' BIOGRAPHIES

Yeming Lu received his BEng degree from Ecole Polytechnique de Montreal in 1999 and MASc degree from the University of British Columbia in 2005. His research interest is in routing algorithms for *ad hoc* networks. Currently, he is a software integration engineer at SR Telecom in St-Laurent, Quebec, Canada.



Vincent W. S. Wong received his BSc degree from the University of Manitoba in 1994, the MASc degree from the University of Waterloo in 1996, and the PhD degree from the University of British Columbia (UBC) in 2000. He worked as a systems engineer at PMC-Sierra Inc. from 2000–2001. He is currently an assistant professor in the Department of Electrical and Computer Engineering at UBC. His current research interests are in resource and mobility management for wireless mesh networks, wireless sensor networks, and heterogeneous wireless networks. Dr Wong received the Natural Sciences and Engineering Research Council postgraduate scholarship and the Fessenden Postgraduate Scholarship from Communications Research Centre, Industry Canada, during his graduate studies. He serves as TPC member in various conferences, including the IEEE ICC and Globecom. He is a member of the IEEE and ACM.

Copyright © 2006 John Wiley & Sons, Ltd.

DOI: 10.1002/dac