www.ietdl.org

Published in IET Communications
Received on 12th August 2011
Revised on 17th September 2012
Accepted on 17th September 2012
doi: 10.1049/iet-com.2011.0543

ISSN 1751-8628

Design of an energy-efficient cross-layer protocol for mobile *ad hoc* networks

Chuan-Chi Weng, Ching-Wen Chen, Po-Yueh Chen, Kuei-Chung Chang

Department of Information Engineering and Computer Science, Feng Chia University, Taichung, Taiwan
E-mail: chingwen@fcu.edu.tw

Abstract: Reducing energy consumption for data transmissions and prolonging network lifetime are crucial in the design of energy-efficient routing protocols. The proportion of successful data transmissions is significant for the reduction of data transmission and traffic load energy consumption, although the energy remaining in node is important for prolonging network lifetime. In this study, the authors propose an energy-efficient cross-layer design for the network layer and medium access control (MAC) layer that reduces energy consumption and prolongs network lifetime. In the network layer, a minimum transmission energy consumption (MTEC) routing protocol is proposed for selecting the MTEC path for data transmission, based on the proportion of successful data transmissions, the number of channel events, the remaining node energy of nodes and the traffic load of nodes. The authors design an adaptive contention window (ACW) for the MAC layer that provides nodes with high successful transmission rates with greater opportunity for contending for a channel to save energy. They used simulations to compare the proposed cross-layer design (MTEC with ACW) with related protocols, including dynamic source routing, traffic-size aware and the Varaprasad routing protocol. The simulation results showed that the proposed cross-layer design (MTEC with ACW) provided better packet delivery rate and throughput than existing protocols. MTEC with ACW also exhibited lower-energy consumption during data transmission and a higher network lifetime than existing protocols.

1 Introduction

Mobile *ad hoc* networks (MANETs) have received much attention in the last decade. Battery power is a significant resource for mobile devices, because most mobile nodes use batteries as the power supply in MANETs. Therefore conservation of energy and energy-efficient routing must be considered when implementing a dynamic and adaptive networking concept for MANETs. Thus, the design of an energy-efficient routing protocol to reduce the energy consumption of data transmissions and prolong network lifetime is an important issue.

Previous energy-efficient routing protocols were divided into two categories, (i) the routing protocols that consider the remaining energy of nodes and the energy consumption of data transmissions [1–8] and (ii) the routing protocols that consider balancing traffic load [9–16]. Based on a general model [17] proposed by Rodoplu and Meng, the transmission distance between two nodes influences the transmission energy consumption. The general model provides a formula for transmission energy consumption $u(d) = d^{\alpha}$, where $u(d)$ is the energy consumption, d is the transmission distance between two nodes and α is a constant between 2 and 4. From this general model, we can see that energy consumption is a direct ratio of the transmission distance raised to the power of α. As a result, the first type of routing protocol is preferred to select a path with minimum transmission energy consumption (MTEC) based on the transmission distance [1–8]. The previous studies [1–8] consider the remaining energy of nodes and the energy consumption of data transmissions to obtain a value of energy cost, where energy cost indicates the value of data transmissions energy consumption divided by the remaining energy of a node. Thus, the path with the minimum energy cost is selected to transmit data. However, considering the energy remaining on nodes can prolong the network, but nodes on the routing path that experience heavy loads may exhaust their energy during data transmission. To take into account traffic load of nodes to prolong the network lifetime, the previous studies discussed various load metrics and proposed several load balance routing protocols [9–16]. These routing protocols combine load-balancing strategies with on-demand route discovery to select the path with the lowest load from all available paths to prolong the network lifetime.

Based on the above description, the design of energy-efficient routing protocol to reduce energy consumption and prolong network lifetime requires consideration of several factors, including the proportion of successful data transmissions, the energy remaining on nodes, and the traffic load of nodes. The use of a contention-based medium access control (MAC) protocol, such as IEEE 802.11 DCF, also means that the number of channel contentions also influences energy consumption. Thus, the design of a MAC protocol to reduce the number of channel contentions is also an important issue when reducing energy consumption. Thus, an energy-efficient protocol should take into account the network layer and the

MAC layer when reducing the energy consumption and prolonging the network lifetime.

In this paper, we propose a cross-layer design for MANETs. For the network layer, we propose a minimum transmission energy consumption routing protocol, called MTEC that finds a path by considering the proportion of successful data transmissions and the traffic load of nodes. For the MAC layer, nodes with highly successful transmission rates have fewer channel contentions to save energy; we use an adaptive contention window (ACW) to dynamically adjust the back-off time. Our proposed cross-layer design can decrease the energy consumption of data transmissions, but it also prolongs network time.

The remainder of this paper is organised as follows. The next section summarises related works on energy-efficient routing protocols. Section 3 proposes a cross-layer protocol with the MTEC routing protocol in the network layer and the ACW design in the MAC layer. Section 4 shows the simulation results. Finally, we conclude with a brief discussion of this work in Section 5.

2 Related work

In this section, we introduce the following two categories of energy-efficient routing protocols: (i) routing protocols [1–8] that consider the remaining energy of nodes and the energy consumption of data transmissions and (ii) routing protocols [9–16] that consider the load of each node when balancing traffic load.

2.1 Routing protocols that consider the remaining energy and data transmission energy

To maximise the network lifetime, some studies [1–8] consider the remaining energy. The minimum battery cost routing (MBCR) [1] protocol considers the remaining energy on nodes to prolong the network lifetime, by selecting the path with the maximum remaining energy from all available paths. To find the path with the maximum remaining energy using MBCR, each node calculates the remaining energy as $f_i(t) = 1/C_i(t)$, where $C_i(t)$ is the energy remaining on node i at time t. MBCR can calculate the sum of the energy remaining for each node in a path using (1), where r_p is a path and l is a link in path r_p. Finally, MBCR selects the path, $B(r)$, with the maximum remaining energy from the set R of all paths, as shown in (2)

$$B(r_\text{p}) = \sum_{i=0}^{l} f_i(t) \tag{1}$$

$$B(r) = \min_{r_\text{p}\in R}\Big(B(r_\text{p})\Big) \tag{2}$$

The min–max battery capacity routing (MMBCR) [2] protocol selects a path where the minimum energy remaining on path nodes is greater than the maximum energy remaining in other paths. MMBCR uses (3), where R is the set of all paths, P is a path and BC_n is the remaining energy of node n.

$$P_\text{MMBCR} = \min_{P\in R}[\max_{n\in P}(1/BC_n)] \tag{3}$$

The condition MMBCR (CMMBCR) [3] protocol considers both the energy consumption during data transmission and the energy remaining on nodes, by combining minimum total transmission power routing (MTPR) [4] protocol with MMBCR. CMMBCR has a predefined threshold for the energy remaining on nodes. When the remaining energy of a node is greater than the threshold, the MTPR protocol is used to reduce energy consumption. However, when the energy remaining on nodes is lower than the threshold, the MMBCR protocol is used to prevent nodes with energy low remaining from being used in the routing path.

Varaprasad proposed a model [5] that simultaneously considered critical node and remaining energy. This model provides a simple equation for computing the lifetime of each node, as shown in (4), where $\text{RP}_i(t)$ and $\text{TP}_i(t)$ are the remaining energy and transmission energy of node i at time t, respectively. In addition, if the system frequently uses critical nodes as intermediate nodes, then these nodes may be drained. Thus, this model also considers critical nodes and provides an equation for measuring the cost of each node, as shown in (5), where $\text{NPR}_i(t)$ is the number of possible paths from the source node to the destination node via node i and $\text{APR}_i(t)$ is the number of possible paths from the source node to the destination node going through node i. According to (5), this model can obtain a weight for a path from the source node to the destination node as shown in (6), where n is the hop count for path k. Therefore this model can select a path using (4) and (6)

$$N_i(t) = \frac{\text{RP}_i(t)}{\text{TP}_i(t)}, \quad \text{where}$$
$$\text{RP}_i(t) = \text{RP}_i(0) - \int_0^t \text{TP}_i(t)\,\text{d}t \tag{4}$$

$$C_i(t) = \frac{\text{NPR}_i(t) - \text{APR}_i(t)}{\text{NPR}_i(t)} \tag{5}$$

$$P_k = \max \sum_{j=0}^{n-1} C_j, \quad \text{where } C_j = C_i(t) \tag{6}$$

2.2 Routing protocols that balance traffic load

Balancing the network traffic load can prolong network lifetime, because nodes with high loads will deplete their batteries quickly, thereby increasing the probability of path breakage or network partitioning. Previous studies discussed various load metrics and proposed several load balance routing protocols [9–16]. These routing protocols combine load-balancing strategies with on-demand route discovery. The path with the lowest load from all available paths leading from the source to the destination is usually selected. These routing protocols can be categorised into three types: (i) delay-based methods that aim to avoid nodes with high link delays, (ii) traffic-based methods that achieve an even distribution of traffic load among network nodes, and (iii) hybrid-based methods combining the features of delay- and traffic-based techniques.

The load-aware on-demand routing (LAOR) protocol [9] is a delay-based load-balancing routing protocol. LAOR achieves load balancing by minimising the estimated total route delay and route hop count.

The associativity-based routing (ABR) protocol [10] is a traffic-based load-balancing routing protocol that adopts associativity to determine the stability of all node neighbours when selecting a path. Associativity requires that each node broadcast a periodic beacon message. A

doi: 10.1049/iet-com.2011.0543

node records the number of beacon messages received from its neighbours in an associativity table. In the path-discovery phase, a node can look up the associativity table to select a stable link on the routing path. Thus, ABR avoids congested nodes. ABR can find a stable routing path, but broadcasting beacon messages causes a loss of performance.

The load-balanced *ad hoc* routing (LBAR) protocol [11] is a traffic-based load-balancing routing protocol. The load metric of LBAR is similar to that of ABR, which is based on active path activity. LBAR considers the activities of neighbours. In the route discovery phase, the source node broadcasts a setup message to its neighbours. After a node receives a setup message, it updates the nodal activity and traffic interference information. Based on this information, the destination node calculates the route load and selects the path with the minimum traffic load.

The traffic-size aware (TSA) protocol [12] is a traffic-based load-balancing routing protocol. In TSA, every intermediate node receiving a path discovery packet calculates its current total load by summing the traffic size at its node and neighbours. The total load is added to the value of the load field of the incoming packet. Thus, TSA selects a path based on the minimum load for transmitting data.

The content sensitive load aware routing (CSLAR) protocol [13] is a hybrid-based load-balancing routing protocol. The route load metric used in CSLAR includes the number of packets in the interface queue, the channel access probability (NAV, network allocation vector), and hop count information. CSLAR uses the information to consider channel contention from neighbours.

3 Energy-efficient cross-layer protocol

This section proposes a cross-layer protocol including the MTEC routing protocol and an ACW to decrease energy consumption during data transmission and prolong the network lifetime. Section 3.1 analyses the behaviour of data transmissions to identify potential factors affecting energy consumption including the proportions of successful data transmissions, the traffic load of nodes, and the number of channel contention events. Based on the derivation in Section 3.1, Section 3.2 proposes an MTEC routing protocol. To improve energy consumption in the MAC layer, Section 3.2 introduces an ACW to dynamically adjust the opportunity of nodes to contend for a channel.

3.1 Analysis of energy consumed during data transmission

This section discusses factors affecting energy consumption during data transmission including the proportion of successful data transmissions, the traffic load of nodes and the number of channel contention events.

Stojmenovic and Lin [18] pointed out that the received signal strength is the inverse ratio of n to the power of the distance, where n is between 2 and 4. Thus, when the distance between two nodes increases, the received signal strength decreases, which results in a decrease in the proportion of successful data transmissions. Thus, we derive the relationship between distance, received signal strength, the proportion of successful data transmissions and energy consumption, as follows.

Assume that the environmental noise is N_0 and the transmitted and received power is P_t and P_r, respectively. Based on the received signal strength and the environmental noise, a receiver can obtain the signal to interference and noise ratio ($\mathrm{SINR_r}$) [19] as shown in (7), where n is a constant between 2 and 4. In addition, the bit error rate (BER) is shown in (8), where $Q(x)$ is the Gaussian probability distribution function [19]. In addition, varied modulation and encoding, for example, BPSK, QPSK and PAM, result in different BER [20]

$$\mathrm{SINR_r} = \frac{P_r}{N_0} = \frac{P_t}{N_0 \cdot d^n} \times \alpha, \quad \text{where } \alpha = \left(\frac{\lambda}{4\pi}\right)^n g_t g_r \tag{7}$$

$$\mathrm{SINR_r} = \frac{P_r}{N_0} = \frac{P_t}{N_0\, d^n} \times \alpha, \quad \text{where } \alpha = \left(\frac{\lambda}{4\pi}\right)^n g_t g_r \tag{8}$$

Using the BER of received data, we can obtain the proportion of successful data transmissions, *SR*, as shown in (9), where E_{PER} is the expectation value of PER (packet error rate), E_{BER} is the expectation value of BER and PS is the packet size. Based on the derivation of these equations, we can see that as the transmission distance between two nodes increases, the proportion of successful data transmissions between these two nodes decreases. Therefore if the proportion of successful data transmissions decreases, the amount of data retransmitted increases and consumes more energy in data transmission.

$$\mathrm{SR} = 1 - E_{\mathrm{PER}}, \quad \text{where} \quad E_{\mathrm{PER}} = 1 - \left(1 - E_{\mathrm{BER}}\right)^{\mathrm{PS}} \tag{9}$$

We next analyse the energy consumption of data transmissions and propose an energy-aware routing metric. We obtain the energy consumption of data transmissions by evaluating the number of data re-transmissions for each link (hop) in a path and the number of channel contention events. In addition, the waiting time for packets remaining in the buffer must also be considered when computing the energy remaining on nodes. Thus, we consider these three factors when computing the energy-aware metric for energy consumption by data transmissions and the energy remaining on the nodes in a routing path.

Assume that link i is a link that connects Node_i to Node_{i+1}. The energy consumed when transmitting a packet j successfully via link i is shown in (10), where β is the number of nodes that contend for the channel in the sender's transmission range. In addition, EC_{rx} and EC_{tx} denote the energy consumed in sending and receiving a packet, respectively, whereas SR_j indicates the probability of the successful transmission of packet j. Therefore after applying (9), the energy consumed in transmitting a packet shown in (10) can be rewritten as (11). As a result, the total energy consumption (TEC) can be calculated using (12), where H is the set of nodes in a path

$$\mathrm{EC}_j = ((\beta - 1) \times \mathrm{EC}_{rx} + \mathrm{EC}_{tx} + \mathrm{EC}_{rx}) \times \left(\frac{1}{\mathrm{SR}_j}\right) \tag{10}$$

doi: 10.1049/iet-com.2011.0543

$$\begin{aligned}\mathrm{EC}_j &= (\mathrm{EC}_{tx} + \beta \times \mathrm{EC}_{rx}) \times \frac{1}{(1 - E_{\mathrm{BER}})^{\mathrm{PS}}} \\ &= (\mathrm{EC}_{tx} + \beta \times \mathrm{EC}_{rx}) \\ &\quad \times \left(1 + \frac{1 - \left(1 - Q\left(C \times d^{(n/2)}\right)\right)^{\mathrm{PS}_j}}{\left(1 - Q\left(C \times d^{(n/2)}\right)\right)^{\mathrm{PS}_j}}\right)\end{aligned} \tag{11}$$

$$\mathrm{TEC} = \sum_{i=1}^{H} \mathrm{EC}_i \tag{12}$$

The traffic load of nodes has to be considered when calculating the energy remaining on nodes in a path after transmission of data packets on a path. To determine the traffic load of nodes, each node keeps a buffer to record the data packets requiring transmission, as shown in Fig. 1. Assume that when the first data packet $P_{k,1}$ of path k arrives at Node$_i$, and there are δ–1 packets in the buffer of Node$_i$; then, the energy consumption of Node$_i$ after transmitting packet $P_{j,1}$ includes transmitting these δ–1 packets in the buffer and packet $P_{k,1}$. Therefore the TEC of Node$_i$ required to transmit these δ–1 packets in the buffer and packet $P_{k,1}$ is shown in (13). Assume that n packets require relaying by Node$_i$ on path k and the remaining energy of Node$_i$ is E_{current}, when the first packet of path j arrives at Node$_i$. The total energy consumption of Node$_i$ is $n \times \mathrm{TEC}$ after transmitting n packets on path k. Therefore the remaining energy ($E_{\mathrm{remaining}}$) of Node$_i$ after transmitting these n packets on path k can be expressed as shown in (14)

$$\mathrm{TEC} = \sum_{j=1}^{\delta} \frac{1}{\mathrm{SR}_j} \times \mathrm{EC}_j \tag{13}$$

$$E_{\mathrm{remaining}} = E_{\mathrm{current}} - n \times \mathrm{TEC} \tag{14}$$

To obtain the relationship between transmission bandwidth and received signal strength, two notebooks with IEEE 802.11b wireless standard were used to transmit data with the ad hoc mode by varying distances where the distance of the two notebooks was set from 5 to 135 m, as shown in Fig. 2. From Fig. 2, we can see that when the distance was less than 20 m, the transmission bandwidth was between 350 and 400 KBps. However, when the distance increased from 21 to 51 m, the transmission bandwidth dropped very fast, from 350 to 150 KBps. When the distance was longer than 50 m, the transmission bandwidth was less than 100 KBps. From the results mentioned above, we can see that the distance between two nodes significantly affected the transmission bandwidth. The main reason is that when the transmission distance increases, the received signal strength decreases to result in the decrease in the successful ratio of data transmissions to affect the transmission bandwidth. Accordingly, a node can use the data in Fig. 2 to determine the potential distance from itself to a sender by received signal strength.

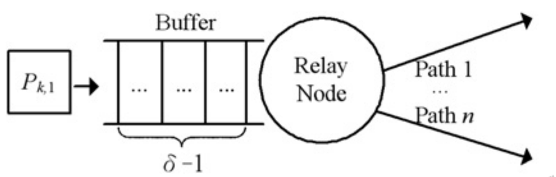

Fig. 1 *Buffer state of relay node*

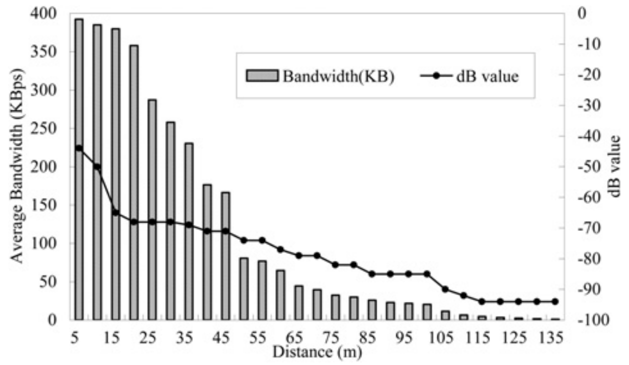

Fig. 2 *Relationships among the distance of nodes, received signal strength and transmission bandwidth*

3.2 Proposed cross-layer protocol

In this section, we propose a cross-layer protocol including the MTEC routing protocol in the network layer and the ACW in the MAC layer. The cross-layer protocol can make the information of the MAC layer be sent to the network layer for routing in MANETs.

3.2.1 MTEC routing protocol in the network layer: MTEC considers the proportion of successful data transmissions, the traffic load of nodes and the number of nodes contending for a channel, to reduce energy consumption and prolong network lifetime. To prolong the network lifetime, MTEC considers the energy remaining on nodes by finding nodes with sufficient energy to completely transmit all data packets. We assume that a source node needs to find a path to transmit N data packets to a destination node and set A indicates the set of paths from the source to the destination. In addition, MRE$_\mathrm{r}$ denotes the minimum remaining energy of the nodes for path r after transmitting N data packets. Let filter MRE (FMRE) be the set of all paths where the value of MRE$_\mathrm{r}$ for path $r > 0$ and $r \in A$, as shown in (15). Thus, if a path belongs to FMRE, each node in the path is active after completely transmitting N data packets. As a result, we can find the path that consumes the minimum energy to transmit N data packets from FMRE, as shown in (16), where r is a routing path from a source to a destination. If FMRE is empty, we select the path that has the minimum energy consumption from set A

$$\mathrm{FMRE} = \{r | \mathrm{MRE_r} > 0, \quad \text{for } dr \in A\} \tag{15}$$

$$\begin{aligned}&\mathrm{Minimum\_Energy\_Consumption} \\ &= \begin{cases} \min\limits_{r \in \mathrm{FMRE}} (\mathrm{TEC}), & \text{if FMRE} \neq \varphi \\ \min\limits_{r \in A} (\mathrm{TEC}), & \text{if FMRE} = \varphi \end{cases}\end{aligned} \tag{16}$$

The route discovery algorithm proceeds as follows. To find the desired path, some fields are added to RREQ (Route Request), including (i) the previous node (PreID); (ii) the current node (CurrentID); (iii) the total data amount (DataSize); (iv) the number of nodes contending for a channel (Current_NCC); (v) the total energy consumption of packets that stored in the buffer (CurrentQ_EC); (vi) the energy remaining on the current node (Current_RE); (vii) the minimum energy remaining on nodes in a routing path (Min_RE); and (viii) the accumulated energy consumption (AEC).

doi: 10.1049/iet-com.2011.0543

When a node receives an RREQ packet during the path discovery phase, the identity of the one-hop neighbour that will become the next relay node is not known. Therefore the potential energy consumed by relaying is calculated by its one-hop neighbour after this node broadcasts RREQ.

One-hop neighbours must calculate energy consumption and energy remaining, so two fields are used in the RREQ, that is, CurrentQ_EC and AEC. We assume the power saving mode (the sleep mode) is not applied in the MAC layer. In addition, with the help of the MAC layer, a node can listen to frames sent by neighbours to know the potential number of active one-hop neighbours. Thus, the node can use the number as the potential number (NCC) of nodes that contend for the channel.

When a node receives an RREQ packet, the value of NCC is written in the Current_NCC field of RREQ. Based on the signal strength provided by the MAC layer, a node can then determine the proportion of successful data transmissions from its one-hop neighbours. After a node receives an RREQ packet, it can determine the NCC value from the previous node and the proportion of successful data transmissions, based on the received signal strength. The node can use these data and (12) to determine the energy consumed by its previous node when transmitting a data packet. Finally, the node received RREQ writes the computed energy consumption in the AEC field of RREQ.

We now determine the energy remaining in nodes after transmitting the DataSize amount of data packets recorded in the RREQ packet. When a node receives an RREQ packet, it can obtain the values of Current_RE and CurrentQ_EC from the received RREQ to determine the previous node's energy remaining and the energy consumed by transmitting the packets in the buffer from the previous node. Therefore the node that received RREQ can use (12) to calculate the energy consumed by transmitting a data packet from the previous node. Thus, once the previous node belongs to the nodes of the routing path and the node receiving the RREQ is the next node in the routing path, the previous node consumes the energy required for relaying a data packet, including transmitting all packets in the buffer (CurrentQ_EC) and a data packet. As a result, the node receiving RREQ can use (14) to compute the energy remaining on the previous node after transmitting a DataSize amount of data. If the energy remaining is less than the MinRE recorded in RREQ, the node updates the MinRE with the computed energy remaining. This node also uses (11) and (12) to calculate the potential energy consumed by transmitting the packets in its buffer, except for the newly received RREQ. Thus, the node writes its energy remaining energy and the computed results in the Current_RE and CurrentQ_EC fields of RREQ, before it broadcasts RREQ.

Finally, when the destination node receives an RREQ packet, it calculates the energy consumption required to transmit a data packet to the previous node and the energy remaining on the previous node after transmitting the DataSize amount of data. In addition, the RREQs received from the destination node can be used to obtain the energy consumption of data transmission on the paths, before selecting the path with the minimum AEC. The destination node replies with a route reply (RREP) packet sent to its one-hop neighbour that sent the selected RREQ packet. The RREP packet is sent back along the path taken by the selected RREQ packet. When the source node receives the RREQ packet, the path with the minimum energy consumption for data transmissions is established. The algorithm of the route request phase is shown in Fig. 3.

With regard to the energy consumption caused by collisions, the IEEE 802.11 DCF protocol can efficiently reduce most of collisions in normal traffic. However, when the traffic is heavy, an ACW can be used to adjust the range of contention window (CW) to reduce collisions.

After the routing path is established, a mobile node in the path sends a HELLO message periodically to inform the

Algorithm:

```
GetPacketInfo(P_i);   //obtaining the related data in the RREQ field from P_i, and receiving
                        signal strength (RSS_i) from node A
RSS_i = B.Received_Signal_Strength;
r = SR(RSS_i) ;   //obtaining successful ratio of data transmissions from Function SR()
                    according to the received signal strength
NCC = P_i.Current_NCC;   //setting the value of NCC that equals to Current_NCC from P_i
n = P_i.DataSize / PacketSize;   //assuming that n is an integer and the value of n is obtained
                                   by rounding up the division
EC_i = GetReqEnergy (r, NCC, n);   //obtaining the required energy for data transmissions
                                     from Function GetReqEnergy() according to the values of r, NCC, and n
EC_in_buffer = P_i.CurrentQ_EC;
RE_A = P_i.Current_RE – (EC_i + EC_in_buffe);
if RE_A < P_i.MinRE then
      P_i.MinRE = RE_A;
end if
P_i.TEC += EC_i;
for q = 1 to Q   //setting the value of Q that equals to the number of packets queued in node B
      P_i.CurrentQ_EC += GetReqEnergy(r_q, NCC_q, n_q);
end for
if B.id == P_i.DestID then
      Cache (P_i);
Else
      P_i.Current_RE = B.RemainingEnergy;
      P_i.CurrentNCC = B.NCC;
      P_i.PreID = P_i.CurrentID;
      P_i.CurrentID = B.ID;
      Broadcast(P_i);
end if
```

Fig. 3 *Algorithm: Node B receiving an RREQ packet P_i from node A*

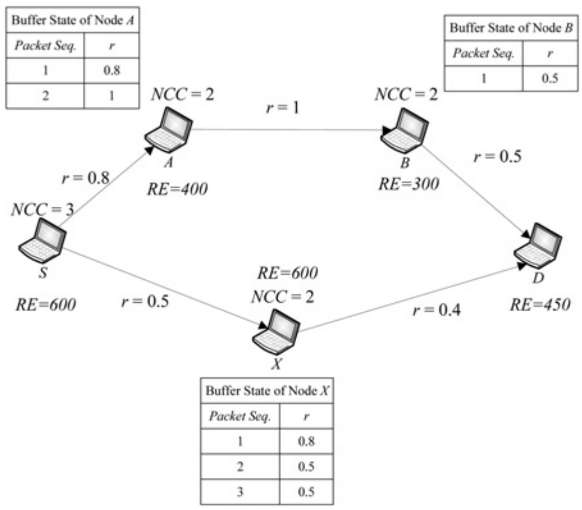

Fig. 4 *Example of discovering the routing path in MTEC*

previous node in the path that it is active. If a node does not receive the HELLO message from the next node in the path, it sends a re-routing message to the source node. When the source node receives a re-routing message, the source node finds a new path to the destination in order to continue transmitting data.

An example is illustrated to explain the design of finding a route by taking into account the following factors including the successful ratio of data transmissions, the number of nodes that contend for the channel, and the number of packets queued in the relay node, as shown in Fig. 4. The number attached in each link is the successful ratio of data transmissions denoted as r, the remaining energy of a node is denoted as RE, and the number of nodes that contend for the channel is denoted as NCC. A table presents the buffer state to record two fields, including the packet sequence number and r. To simplify the calculation of energy consumption for receiving and sending a packet, we assume that the energy consumption for receiving and sending a packet is one and four energy units, respectively.

Assume that source node S wants to send ten packets to destination node D and there are two paths S–A–B–D and S–X–D for choice. The energy consumption of transmitting a packet from a sender to a receiver includes the energy consumed by overhearing the transmissions of neighbours when the sender does not obtain the right of using the channel, transmitting all packets in the buffer of the sender, and transmitting the packet from the sender to the receiver. Therefore, for path S–A–B–D in Fig. 4, node S consumed 7.5 energy units to send a packet to node A where 7.5 is calculated by $((\mathrm{NCC_S}-1)\times \mathrm{EC}_{rx}+\mathrm{EC}_{tx})\times(1/r_{\mathrm{S-A}}) = ((3-1)\times 1+4)\times(1/0.8)$ and $\mathrm{NCC_S}$ and $r_{\mathrm{S-A}}$ indicate NCC of node S and the successful rate of data transmissions of link SA, respectively. The remaining energy (RE) of node S after transmitting ten packets is $\mathrm{RE}-10\times((\mathrm{NCC_S}-1)\times \mathrm{EC}_{rx}+\mathrm{EC}_{tx})\times(1/r_{\mathrm{S-A}}) = 600-10\times((3-1)\times 1+4)\times(1/0.8) = 525$. Node A consumed $((\mathrm{NCC_A}-1)\times \mathrm{EC}_{rx}+\mathrm{EC}_{tx}+\mathrm{EC}_{rx})\times(1/r_{\mathrm{A-B}}) = ((\mathrm{NCC_A}-1)\times 1+4+1)\times(1/0.8+1/1+1/1) = 19.5$ energy units for receiving the packet and transmitting the packets in the buffer and the new coming packet. The remaining energy of node A after transmitting ten packets is $\mathrm{RE}-10\times((\mathrm{NCC_A}-1)\times \mathrm{EC}_{rx}+\mathrm{EC}_{tx}+\mathrm{EC}_{rx})\times(1/r_{\mathrm{A-B}}) = 400-10\times((2-1)\times 1+4+1)\times(1/0.8+1/1+1/1) = 205$ energy units. In addition, node B consumed $((\mathrm{NCC_B}-1)\times 1+4+1)\times(1/0.5+1/0.5) = 24$ energy units for transmitting a packet. The remaining energy of node B is $\mathrm{RE}-10\times((\mathrm{NCC_B}-1)\times \mathrm{EC}_{rx}+\mathrm{EC}_{tx}+\mathrm{EC}_{rx})\times(1/r_{\mathrm{S-A}}) = 300-10\times((2-1)\times 1+4)\times(1/0.5+1/0.5) = 60$ energy units. Finally, node D consumed 1 energy unit to receive a packet. Therefore after ten packets are sent from node S via path S–A–B–D to node D, the total energy consumed is 400 energy units. The $\mathrm{MRE_{S-A-B-D}}$ (minimum remaining energy) via path S–A–B–D is 60 energy units which are larger than zero. With regard to transmitting a packet through path S–X–D, 57.5 energy units are consumed where 10 energy units are consumed by node S to send the packet to node X, node X consumes 46.5 energy units $(((\mathrm{NCC_X}-1)\times 1+4+1)\times(1/0.8+1/0.5+1/0.5+1/0.4) = 46.5)$ to receive the packet and send all packets in the buffer and the new coming packet, and node D consumes 1 energy unit to receive the packet. After transmitting ten packets, the total consumption of path S–X–D is 575 energy units. The $\mathrm{MRE_{S-X-D}}$ (minimum remaining energy) is 135 energy units which are larger than zero. As paths S–A–B–D and S–X–D consume 412.5 and 575 energy units and the remaining energy of the nodes in the two routes after transmitting ten packets are larger than zero. Therefore our proposed MTEC selects path S–A–B–D which has lower-energy consumption for data transmissions to transmit data.

3.2.2 ACW design in the MAC layer: This section proposes the design of an ACW in the MAC layer to reduce energy consumption and increase throughput. Based on the derivation given Section 3.1, it can be seen that the received signal strength affects the proportion of successful data transmissions and energy consumption. Previous research proposed a multi-rate protocol to improve the proportion of successful data transmissions. However, if a lower transmission rate is used to transmit data, the transmission time increases and consumes more energy. Furthermore, the performance anomaly problem identified by Heusses *et al.* [21] applies to multi-rate MAC protocols. Thus, the overall network throughput tends towards the throughput with the lowest transmission rate. This is because the probability of using the channel for each node is approximately the same, so the link using a lower transmission rate takes more transmission time than one using a higher transmission rate. As a result, links with a lower transmission rate or a low proportion of successful data transmissions result in increased transmission time and greater energy consumption.

To improve network throughput and reduce the energy consumption of data transmissions, we modified the IEEE 802.11 DCF protocol to use an ACW. Based on the proportion of successful data transmissions, a node uses an ACW to dynamically adjust the back-off time between different nodes. ACW allows links with higher proportion of successful data transmissions to have smaller ranges of CWs and a greater probability of using the channel. Thus, links with higher proportion of successful transmission will have greater opportunity to use the channel, which increases throughput and reduces the energy consumption by reducing the amount of re-transmissions.

We assume that a node can listen to control frames and data frames from its one-hop neighbours to determine the proportion of successful transmissions between neighbours and the node, based on the received signal strength. Therefore we divided the range of CW into k levels, where the range of the CW for the nth level is nW. If there are n nodes contending for a channel and the ranges of their CW of these n nodes are W, $2W$, …, kW, the probability of using the channel for these k nodes can be represented as (17), where $P(iW)$ is the probability of using the channel and iW is the range of CW. For example, if there are three

doi: 10.1049/iet-com.2011.0543

nodes (nodes A, B and C) and their ranges of CW are 10W, 20W and 30W, the probability of these three nodes to use the channel is 1/10:1/20:1/30 = 6:3:2

$$P(W):P(2W):P(3W):\cdots:P(kW) = \frac{1}{1}:\frac{1}{2}:\frac{1}{3}:\cdots:\frac{1}{k} \quad (17)$$

The binary exponential back-off (BEB) algorithm [22] is popularly used in the MAC layer to reduce collisions because of small CW. However, the BEB algorithm has two problems. (i) BEB sets double back-off time when a collision occurs and sets back-off time to minimum (CW_{min}) when a transmission succeeds. Therefore the fairness is lack in the BEB algorithm. (ii) In BEB, the range of CW is adjusted after a collision or a successful transmission occurs, so it incurs unnecessary adjustment.

To provide fairness and to solve collisions for heavy traffic in the MAC layer, in our proposed ACW, a new parameter, CW_{TH}, is added to play the role of a threshold for distinguishing heavy traffic load from light traffic load. The back-off algorithm for ACW can be represented as follows where CW_{min} and CW_{max} indicate the minimum and the maximum of CW

CW == Max(CW/2, CW_{TH}), if (succeeds and CW > CW_{TH}).
CW == Max(CW/2, CW_{min}), if (succeeds and CW <= CW_{TH}).
CW == Min(2 × Max(CW, CW_{TH}), CW_{max}), if a collision occurs.
CW == CW, if the number of retrying limitation is reached.

where

$$0 <= CW_{min} <= CW <= CW_{max}$$

$$0 <= CW_{min} <= CW_{TH} <= CW_{max}$$

In the above example, the CW_{max} is $50W$, CW_{min} is $5W$ and CW_{TH} is $20W$, when the *CW* of node A is $10W$, the size of CW of node A is set to $5W$ after node A successfully transmits a packet. If the CW_{min} is $8W$, the size of CW of node A becomes $8W$. However, if a collision occurs and the size of CW of node *A* is $10W$, the size of CWs of node A is set to Min(2 × Max(CW, CW_{TH}), CW_{max}) = Min(2 × Max ($10W$, $20W$), $50W$) = $40W$.

The proposed algorithm can avoid the fast adjustment of the range of CW. With regard to the fairness problem, our proposed ACW promotes the nodes that do not contend for the channel by shortening their CW to increase the probability of using the channel.

In addition, to avoid collisions in heavy traffic, the CW can be adjusted depending on the number of neighbours that want to contend for the channel. When the traffic load is light, the increase of collisions occurred by the division of CW into k levels is very slight. However, when the traffic load is heavy, the collisions may increase because of small CW. To solve this problem, a node can set up the CW by taking into account the number of potential active neighbours. That is, the CW of a node is equal to the sum of its original CW and $c\beta$ where c is a constant and β is the number of potential active neighbours. With the design of ACW, the collisions can be reduced when the traffic load is heavy.

4 Simulation results

This section describes the simulation environments and presents the simulation results. The NS2 simulator [23] (version 2.34) was used to simulate dynamic source routing (DSR) [24], Varaprasad routing protocol, TSA [12] and our proposed MTEC with/without ACW. The contention-based IEEE 802.11 DCF protocol was adopted as the protocol used in the MAC layer in these other routing protocols.

The simulation environments and parameters were as follows. The random way-point (RWP) model was used to evaluate the influence of node mobility on throughput and energy consumption by the protocols. All mobile nodes were distributed in an area of $1000 \times 1000\ m^2$, and the speed of the nodes was set at 030 m/s. A node started a trip to a random destination at a constant speed chosen from a uniform distribution (between 0 and 30 m/s). The source and destination patterns were generated using the CBR traffic model. Each node initially had 100 J. The simulation results were obtained by averaging 100 simulated results for each simulation. The formula for the data generation rate is shown in (18), where PS is the packet size and TI is the time interval for transmitting a packet. For example, if PS was 1500 bytes and the data generation rate was 0.6 Mbps, the value of TI would be 0.02 s. In addition, we also wrote a function in NS2 to transfer the received signal strength into the distance by referring to Fig. 2. With regard to noise, in addition to the environmental noise provided by NS2, the interference noise among nodes, N_0, which is appeared in (7)–(9) was also considered in NS2. The detailed parameters are listed in Table 1.

IEEE 802.11b permits several PHY levels, including 11, 5.5, 2 and 1 Mbps. A node determines the data transmission rate by comparing the SNR value with the SNR thresholds

Table 1 Simulation parameters

Parameters	Values
topology size	1000 m × 1000 m
simulation time	600 s
number of nodes	100
radio propagation model	two ray ground
MAC type	IEEE 802.11 DCF
queue size	100
traffic model	CBR
WLAN standard	802.11b
radio propagation range	250 m
transmission rate for data signal	11, 5.5, 2, and 1 Mbps
transmission rate for control signal	1 Mbps
data size	2 Mbytes/flow
packet size	1500 bytes
data generation rate	(0.2/0.4/0.6/0.8/1.0) Mbps
mobility model	random way-point (RWP)
mobility velocity	0–30 m/s
P_t value	0.28183815 W

Table 2 Relationship between transmission rate, transmission range and signal strength

Transmission rate, Mbps	Transmission range, m	Signal strength, W
11	0–123.57	1.82644×10^{-9}
5.5	123.57–138.80	1.44745×10^{-9}
2	138.80–228.13	5.2675×10^{-10}
1	228.13–295.76	1.86468×10^{-10}

Table 3 Energy consumption of different radio modes with IEEE 802.11b Lucent wave LAN

Radio mode	Energy consumption (EC)
point-to-point send	1.9 × size μWs/byte + 420 μWs
point-to-point receive	0.42 × size μWs/byte + 330 μWs
broadcast send	1.9 × size μWs/byte + 250 μWs
broadcast receive	0.42 × size μWs/byte + 56 μWs

stated in the ARF protocol [25] as shown in Table 2. The energy consumed in broadcasting, receiving and transmitting data can be found by reference to the paper [7, 8], where the authors used the Lucent wave LAN wireless adapter and the IEEE 802.11b protocol to measure energy consumption, as shown in Table 3. In our simulation, the energy consumption of transmitting, receiving or broadcasting a packet for discovering the routing path and data transmissions refers to Table 3

$$\text{rate} = \frac{\text{PS} \times 8}{\text{TI}} \tag{18}$$

The simulation results that were compared for MTEC and these three related works in terms of average end-to-end delay, packet delivery ratio, number of packet timeouts, throughput; the energy consumption during data transmissions, the energy consumption of packet transmission, the energy efficiency and the network lifetime (survival nodes ratio).

Fig. 5 shows the average end-to-end delay of DSR, TSA, Varaprasad and MTEC. As MTEC shows lower queuing delay, contention delay and transmission delay than other routing protocols, thus, MTEC had a lower average end-to-end delay than other routing protocols, as shown in Fig. 5. When IEEE 802.11 was used in the MAC layer, MTEC had a lower average end-to-end delay than Varaprasad, TSA and DSR by 23.8–33.7%. When ACW was used in MTEC, the average end-to-end delay was lower than Varaprasad, TSA and DSR by 74.3–77.7%.

Fig. 6 shows the packet delivery ratios with different data generation rates. Fig. 6 shows MTEC with IEEE 802.11 and MTEC with ACW provided higher packet delivery ratios than other routing protocols by 15.85 and 23.84%, respectively. Fig. 7 shows the number of timeout packets in the TOUT (DROP_RTR_QTIMEOUT) field obtained by

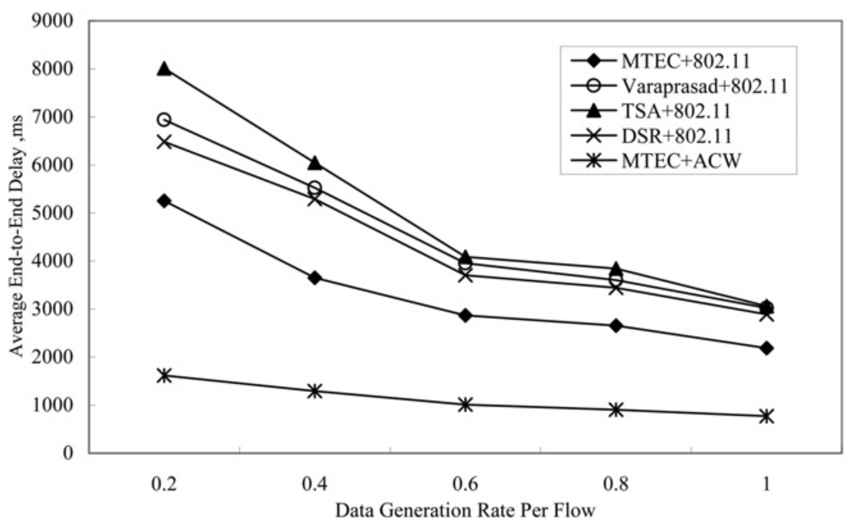

Fig. 5 *Average end-to-end delay*

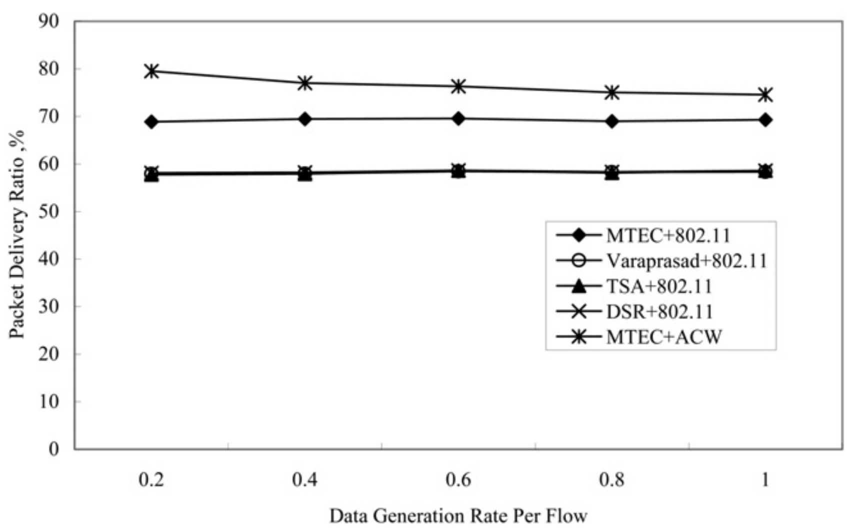

Fig. 6 *Packet delivery ratio*

doi: 10.1049/iet-com.2011.0543

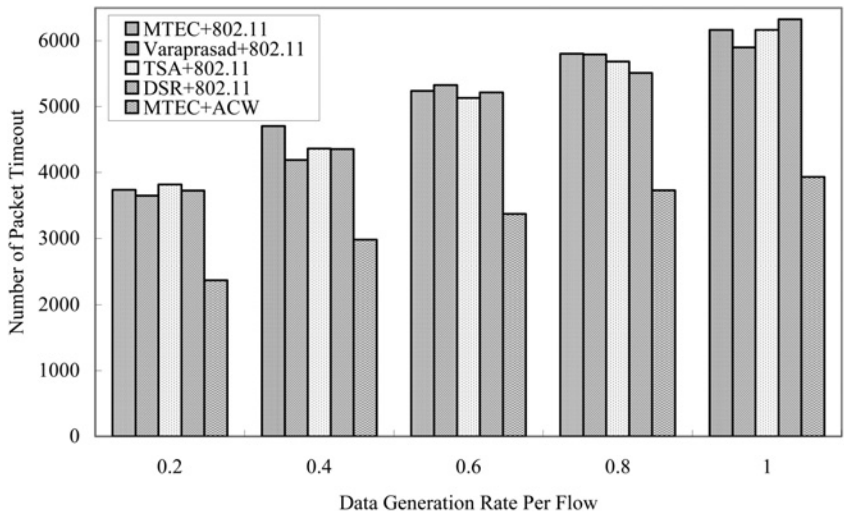

Fig. 7 *Number of packet timeouts*

the NS2 simulator. Fig. 7 shows that MTEC with ACW delivered the lowest number of timeout packets among all routing protocols. This is because ACW dynamically adjusts the CW range based on the proportion of successful data transmissions, which increases the proportion of packets successfully delivered.

Based on the analysis and discussion above, we can see that MTEC produced a lower average end-to-end delay and a higher delivery ratio of packets than the other routing protocols. Therefore MTEC provided better throughput than the other methods, as shown in Fig. 8. From Fig. 8, we can see that the throughput of MTEC with IEEE 802.11 was 0.24 Mbps, which was 2.26 times better than Varaprasad, TSA and DSR, on average. The throughput of MTEC with ACW was 0.57 Mbps, which was 5.51 times better than Varaprasad, TSA and DSR, on average.

After the path discovery phase, the required path was selected for data transmission. The average energy consumption when transmitting a data packet is shown in Fig. 9. Fig. 9 shows that MTEC had the lowest average energy consumption. This was because MTEC considered the proportion of successful data transmissions, which reduced the time of re-transmitting data packets, and the number of channel contention events, which decreased the energy consumption when listening to neighbours' transmissions. MTEC with IEEE 802.11 consumed less energy than Varaprasad, TSA and DSR by 19.50–27.77%. The ACW method with MTEC consumed even less energy than Varaprasad, TSA and DSR by 83.52–85.22%.

Fig. 10 shows the energy efficiency, that is, the number of successful data packets received by the destination per joule, whereas the energy consumption included the energy consumed by discovering and re-routing the required path and the energy consumed by transmitting data packets. Fig. 10 shows that MTEC with IEEE 802.11 successfully received 1.84 packets, which was 1.86 times better than Varaprasad, TSA and DSR, on average. MTEC with ACW successfully received 7.41 packets, which was 7.46 times better than Varaprasad, TSA and DSR, on average.

Fig. 11 shows the proportion of nodes that survived during the network lifetime. We determined the survival of nodes in the network, by adding one source and one destination located at (200, 200) and (800, 800) in the network, respectively. The sender transmitted data via multiple hops to the destination at a data generation rate of 0.6. The simulation time was 500 s. Fig. 11 shows that Varaprasad considered the energy remaining and delivered a higher nodes survival rate than

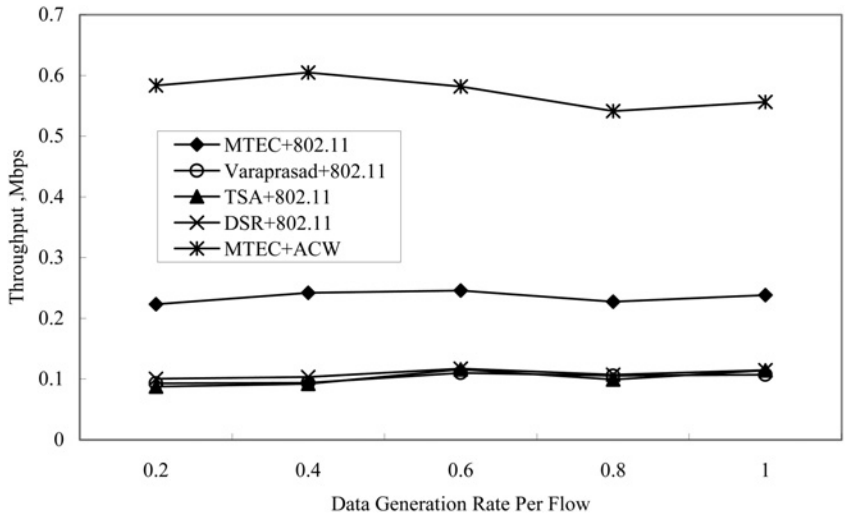

Fig. 8 *Throughput*

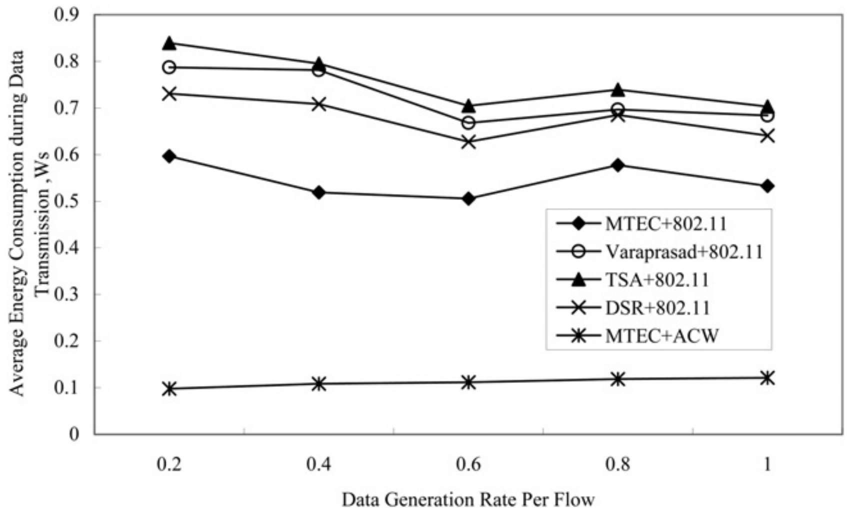

Fig. 9 *Average energy consumption during data transmission*

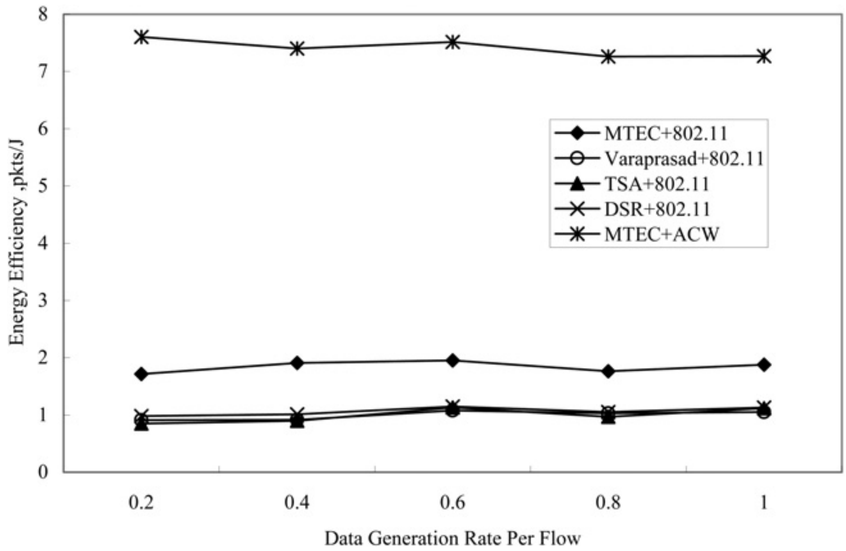

Fig. 10 *Energy efficiency*

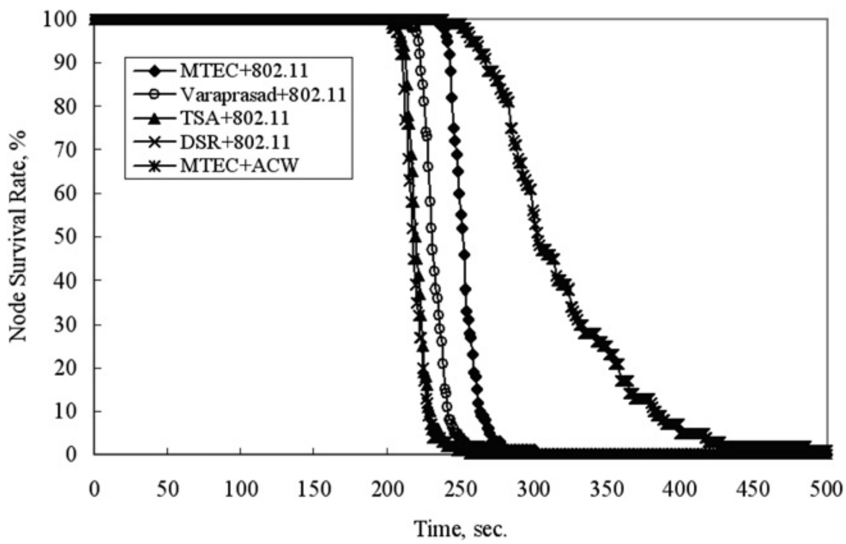

Fig. 11 *Node survival rate*

IET Commun., 2013, Vol. 7, Iss. 3, pp. 217–228
doi: 10.1049/iet-com.2011.0543

DSR or TSA. MTEC with IEEE 802.11 considered the energy consumed by transmitting data and the energy remaining on nodes, which provided a higher survival rate than Varaprasad, TSA or DSR. MTEC with ACW produced a higher survival rate than MTEC with IEEE 802.11. This was because ACW ensures a higher proportion of successful data transmissions by giving a higher probability of transmitting data. MTEC with ACW efficiently decreased the number of data packets re-transmitted, which reduced energy consumption.

From the simulation results, we can see that when all protocols use the IEEE 802.11 DCF protocol in the MAC layer, DSR shows shorter end-to-end delay than TSA and Varaprasad because DSR uses the route with minimal hop counts to transmit data. However, because DSR does not take into account the remaining energy of nodes and the network traffic, it shows a lower node survival rate than TSA and Varaprasad. TSA selects the route with minimal traffic to transmit data, so the route selected by TSA may be longer than others to result in more energy consumption than others. In addition, TSA does not consider the remaining energy of nodes, so the node survival rate in TSA is lower than those in METC and Varaprasad. Varaprasad takes into account the remaining energy of nodes and transmission energy to select the routing path, so it shows a higher node survival rate than DSR and TSA. However, Varaprasad does not consider the successful rate of data transmissions, so it consumes more energy for transmitting a data packet than MTEC. Therefore Varaprasad shows a lower node survival rate than MTEC.

Rather than DSR, TSA and Varaprasad, our proposed MTEC takes into account the factors including the proportion of successful data transmissions, the traffic load of nodes, the number of nodes contending for a channel and energy remaining of nodes to reduce the energy consumption for data transmissions and prolong network survival time. Therefore the routing path selected by MTEC shows better throughput and lower-energy consumption for data transmissions than other protocols.

In addition, when MTEC adopts the ACW protocol in the MAC layer, from the simulation results, we can see that the energy consumption of transmitting data packets can be reduced further and the node survival rate is also prolonged. The main reason is that ACW can make a communication pair with high successful rate to have high probability to obtain the right of using the channel.

To select the desired path, TSA, Varaprasad and our MTEC need to wait for a period of time to receive RREQs at a destination after the destination receives the first RREQ. The overhead of discovering the desired path for these three protocols is higher than DSR. For example, TSA selects the path with minimal traffic from all collected paths. Varaprasad selects the path from all collected paths by considering the remaining energy of nodes. With regard to our proposed MTEC, the path is selected from all collected paths by taking into account the successful rate of data transmissions, the traffic load of nodes and the number of channel contentions. Therefore the overhead of MTEC is higher than others in discovering the desired path. Although MTEC pays for the overhead in the path discovery phase by taking into these factors, MTEC have shorter end-to-end delay, a higher packet delivery rate, higher throughput and lower-energy consumption for data transmissions than other protocols because MTEC takes these factors into account.

5 Conclusions

This paper proposed a cross-layer protocol to increase throughput and decrease energy consumption during data transmissions. In the network layer, we considered the proportion of successful data transmissions, the number channel contention events and the number of packets remaining in a node's queue in our proposed MTEC routing protocol. In the MAC layer, we proposed an ACW design that dynamically adjusted the range of the CW based on the proportion of successful data transmissions, which improved network throughput and energy consumption.

The contributions of this paper are as follows. First, the proportion of successful data transmissions, the traffic load of nodes, and channel contentions in the MAC layer were considered in the design of our energy-efficient MTEC routing protocol, which efficiently decreased energy consumption during data transmission and prolonged the network lifetime. Second, the routing path with a high proportion of successful data transmissions was given a higher probability of using the channel, and we adjusted the range of the CW based on the proportion of successful data transmissions. The simulation results show that our proposed cross-layer protocol design (MTEC with ACW) had 5.8 times the throughput and a higher packet delivery ratio than Varaprasad, TSA and DSR (22.23% more on average). In addition, the energy efficiency of MTEC with ACW was much higher than Varaprasad, TSA and DSR (5.7 times higher on average).

6 Acknowledgment

This research was supported by the National Science Council of the Republic of China, Taiwan, under grant no. NSC 100-2221-E-035–069 and NSC101-2221-E-035-067.

7 References

1 Singh, S., Woo, M., Raghavendra, C.S.: 'Power-aware routing in mobile ad hoc networks'. Proc. Fourth Annual ACM/IEEE Int. Conf. Mobile Computing and Networking (MobiCom'98), Dallas, Texas, October 1998, pp. 181–190

2 Toh, C.-K.: 'Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks', *IEEE Commun. Mag.*, 2001, **39**, pp. 138–147

3 Toh, C.-K.: 'Performance evaluation of battery-life-aware routing schemes for wireless ad hoc networks'. Proc. IEEE Int. Conf. Communications (ICC 2001), Helsinki, Finland, June 2001, vol. 9, pp. 2824–2829

4 Kim, K.-R., Min, S.-G., Yu, N.-K.: 'Maximizing the lifetime of wireless ad hoc networks using minimizing the maximum used power routing'. Proc. Third Int. Symp. Wireless Pervasive Computing (ISWPC 2008), Santorini, Greece, May 2008, pp. 557–561

5 Varaprasad, G.: 'Lifetime enhancement routing algorithm for mobile ad hoc networks', *IET Commun.*, 2011, **5**, (1), pp. 119–125

6 Ferrari, G., Malvassori, S.A., Tonguz, O.K.: 'On physical layer-oriented routing with power control in ad hoc wireless networks', *IET Commun.*, 2008, **2**, (2), pp. 306–319

7 Weng, C.-C., Chen, C.-W., Ku, C.-J., Hwang, S.-F.: 'A bandwidth-based power-aware routing protocol with low route discovery overhead in mobile ad hoc networks', *Comput. J.*, 2010, **53**, pp. 969–990

8 Chen, C.-W., Weng, C.-C., Kuo, Y.-C.: 'Signal strength based routing for power saving in mobile ad hoc networks', *J. Syst. Softw.*, 2010, **83**, (8), pp. 1373–1386

9 Song, J.-H., Wong, V., Leung, V.: 'Load-aware on-demand routing (LAOR) protocol for mobile ad hoc networks'. Proc. 57th IEEE Semi. Vehicular Technology Conf. 2003 (VTC 2003-Spring), Jeju, Korea, April 2003, pp. 1753–1757

10 Toh, C.-K.: 'Associativity-based routing for ad hoc mobile networks', *Wirel. Pers. Commun.*, 1997, **4**, (2), pp. 1–36

11 Hassanein, H., Zhou, A.: 'Routing with load balancing in wireless ad hoc networks'. Proc. Fourth ACM Int. Workshop Modeling Analysis and Simulation of Wireless and Mobile Systems (MSWiM'01), Rome, Italy, July 2011, pp. 89–96
12 Altalhi, A.H., Richard, G.III: 'Load-balanced routing through virtual paths: highly adaptive and efficient routing scheme for ad hoc wireless networks'. Proc. 23rd IEEE Int. Performance Computing and Communications Conf. (IPCCC 2004), Phoenix, Arizona, April 2004, pp. 407–413
13 Li, Y., Man, H.: 'Three load metrics for routing in ad hoc networks'. Proc. 60th IEEE Vehicular Technology Conf. (VTC 2004-Fall), Los Angeles, CA, September 2004, vol. 4, pp. 2764–2768
14 Adibi, S., Agnew, G.B.: 'Multilayer flavoured dynamic source routing in mobile ad-hoc networks', *IET Commun.*, 2008, **2**, (5), pp. 690–707
15 Chang, C.-Y., Wang, Y.-P., Chao, H.-C.: 'An efficient mesh-based core multicast routing protocol on MANETs', *J. Internet Technol.*, 2007, **8**, (2), pp. 229–239
16 Yen, Y.-S., Chao, H.-C., Chang, R.-S., Vasilakos, A.: 'Flooding-limited and multi-constrained QOS multicast routing based on the genetic algorithm for MANETs', *Math. Comput. Model.*, 2011, **53**, (11–12), pp. 2238–2250
17 Rodoplu, V., Meng, T.H.: 'Minimum energy mobile wireless networks', *IEEE J. Sel. Areas Commun.*, 1999, **17**, pp. 1333–1344
18 Stojmenovic, I., Lin, X.: 'Power aware localized routing in wireless networks', *IEEE Trans. Parallel Distrib. Syst.*, 2001, **12**, (11), pp. 1122–1133
19 Rappaport, T.: 'Wireless communications: principles and practice' (Prentice-Hall PTR, 2002, 2nd edn.)
20 Proakis, J.G., Salehi, M.: 'Fundamentals of communication systems' (Prentice Hall, 2004)
21 Heusse, M., Starzetz, P., Rousseau, F., Berger-Sabbatel, G., Duda, A.: 'Performance anomaly of 802.11b'. Proc. 22nd Annu. Joint Conf. IEEE Computer Communication Societies (INFOCOM 2003), San Franciso, CA, USA, March–April 2003, vol. 2, pp. 836–843
22 IEEE 802.11, 'Wireless LAN medium access control (MAC) and physical layer (PHY) specifications', 1999
23 The Network Simulator - NS2, http://www.isi.edu/nsnam/ns/
24 Johnson, D.B., Maltz, D.A.: 'Dynamic source routing in ad-hoc wireless networks', in: Imielinski, T., Korth, H. (Eds.): 'Mobile computing' (Kluwer Academic Publishers, 1996), Ch. 5, pp. 153–181
25 Kamerman, A., Monteban, L.: 'WaveLAN®-II: a high-performance wireless LAN for the unlicensed band', *Bell Labs Tech. J.*, 1997, **2**, (3), pp. 118–133

doi: 10.1049/iet-com.2011.0543