Wireless Personal Communications (2019) 107:2191–2209
https://doi.org/10.1007/s11277-019-06378-y

# Energy and Velocity Based Tree Multicast Routing in Mobile Ad-Hoc Networks

**Abu Sufian**[1] · **Anuradha Banerjee**[2] · **Paramartha Dutta**[3]

## Abstract

Nodes in ad hoc networks are mostly mobile, moving with arbitrary velocity and direction. Therefore, it is always beneficial if nodes are equipped with alternative paths to successors. The present article proposes both energy and velocity based (together called weight based) tree multicast protocol where not only the relative velocity of nodes, but also their expected residual lifetime contribute in computing acceptability of a path from a multicast sender to a multicast receiver. Experimental results confirm superiority of the proposed scheme over existing state-of-the-art multicast protocols, in terms of data packet delivery ratio, multicast route lifetime, control message overhead and end-to-end delay.

**Keywords** Ad hoc networks · Energy efficient · Multicasting in MANETs · Velocity of node · Weight of path · Tree routing

## 1 Introduction

An ad hoc network is an interconnection of autonomous mobile nodes which capable of sending and receiving radio signals that move with arbitrary velocity in a random direction. No existing infrastructure or centralized administration is required [1, 2]. That's why this kind of networks is very useful in emergency situations like communication in war, after natural disaster, aircraft and marine communication, industrial and other scenarios.

A number of multicast routing protocols have come to existence in the literature of ad hoc networks [3–9]. Some of these protocols are based on the concept that mobility parameters are fixed whereas the others consider mobility from various perspectives. Clearly, the protocols that consider inherent dynamism of ad hoc networks outperform the ones based on fixed parameters. The present approach, energy and velocity (together called 'Weight') based tree multicast routing or simply WTMR incorporates the idea of expected residual

---

✉ Abu Sufian
sufian.csa@gmail.com

[1] Department of Computer Science, University of Gour Banga, Malda, WB, India

[2] Department of Computer Applications, Kalyani Government Engieering College, Kalyani, WB, India

[3] Department of Computer and System Sciences, Visva-Bharati University, Shantiniketan, WB, India

lifetime and velocity of nodes. A node with larger residual energy cannot be guaranteed to last longer than another node with smaller residual energy. Energy depletion rate plays an important role there. Similarly velocity of a node also very crucial. Priority is given to the links that have alternatives with lifetime not less than the lifetime of the original link. These alternatives play a great part in reducing control message overhead. Message packets can be forwarded through them in case of breakage of the direct link. Less control message yields less energy consumption. As a result, packet delivery ratio increases significantly.

The rest of the paper organized as follows: in Sect. 2 related works has studied, Sect. 3 explain the present strategy that is WTMR, optimum route selection has discussed in Sect. 4, in Sect. 5 computation of weight has explained, in Sect. 6 discussion of simulation results and conclusion has drawn in Sect. 7.

## 2 Related Work

Among conventional multicast routing protocols, Multicast On-demand Distance Vector Routing (MAODV) [10], On-demand Multicast Routing protocol [11], Ad Hoc Multicast Routing (AMRoute) [12] are mention-worthy. In MAODV, whenever a route-request (RREQ) packet reaches at a multicast destination, it sends back one route-reply (RREP) packet. Forwarding paths are created by storing identifier of a immediate predecessor in the routing table at each node. In ODMRP [11], the multicast sender floods JOIN_REQ throughout the network. As soon as the message arrives at a non-destination router, it stores multicast session id, sender id and immediate predecessor in a table before further propagating it. On the other hand, whenever a multicast destination receives the JOIN_REQ, it does not forward that any further and sends back a JOIN_REPLY message to its immediate predecessor. Components of JOIN_REPLY are multicast session id and sender id. As the JOIN_REPLY reaches an ordinary node, it forwards that to immediate predecessor mentioned in the routing table provided multicast session ids and sender ids received in JOIN_REPLY matches with an earlier JOIN_REQ. in that way, JOIN_REPLY ultimately reaches the multicast sender and the sender is now able to send data packets through an established route. AMRoute [12] is a tree-based protocol that relies on the underlying unicast routing protocol. Physically close multicast members construct proactive bi-directional tunnels connecting each pair. These tunnels form a mesh. A multicast tree is created from multicast source to the core of each group. In DLBMRP [13] authors proposed a multicast protocol where traffic load are balanced. Here they first classified all the nodes into three random groups then designed tree for each group to efficiently data transfers.

Several power-aware multicast protocols have already evolved in literature [14–19]. Power aware routing in mobile ad hoc networks [20], Scalable energy efficient Location Aware Multicast protocol (SEELAMP) [21], Energy Efficient Clustering Technique (EECT) [22], FESC [23] Energy Efficient Multicasting based on Genetic Algorithm [24], Power-aware Multicast Routing (PAMR) [25], Energy Efficient Multicast Routing (EEMR) [17] etc. are important. In [20] authors discussed the importance of power-aware routing protocols. They used use $cost / packet$ and $maximumnodecost$ as metrics rather than traditional metrics such as $hopcount$ or $delay$. SEELAMP [21] adopts the concept of zone routing. The network is divided into certain zones and one head node keeps track of latitude and longitude of all other nodes in the network. It has the responsibility of maintaining stable connectivity with all other nodes belonging to the same zone. Bi-directional multicast trees are created within each zone to reduce message consumption in the network. EECT

[22] is an energy efficient clustering technique where energy efficient clusters are formed based on transmission power, residual energy, and velocity. Within each energy-efficient cluster, MAODV is implemented for multicasting. In FESC author proposed a single hop clustering scheme. Here more powerful and the less mobile node would be clustered head and other nodes within radio range directly attach to cluster head. In this scheme more stable and energy efficient routing could be possible as other nodes just sit idle to save energy.

# 3 WTMR in Detail

## 3.1 Why Only Residual Energy is Not Sufficient

The intention behind electing a route with a maximum of minimum battery power is the expectation that this route will survive for a long time. But that may not be true in actual scenario. For example, consider two nodes $n_i$ and $n_j$ with their residual powers being 100 mj and 120 mj, respectively while the highest possible battery power are 500 mj and 400 mj respectively. According to the study of discharge curve of batteries heavily used in ad hoc networks, at least 40% of total battery power is required to remain in operable condition [7]. Therefore, the minimum possible values of residual powers of $n_i$ and $n_j$ are 200 mj and 160 mj.

Maximum battery power that can be consumed yet, by $n_i$, is (200–100) mj i.e. 100 mj, while the same of $n_j$ is (160–120) mj i.e. 40 mj. Assuming energy depletion rates of $n_i$ and $n_j$ are 5 mj/ms and 4 mj/ms. Therefore, $n_i$ will survive for (100/5) ms i.e. 20 ms while $n_j$ will serve for (40/4) ms i.e. 10 ms. So, a node with higher residual energy does not necessarily live longer.

## 3.2 Network Model

Initially, the network is modeled as a graph $G = (V, E)$, where $V$ is the set of vertices and $E$ is the set of edges. WTMR extracts $G'$ from $G$ such that $G' = (V', e(s, \alpha(s)));\alpha(s) \in V'$ where $V'$ is the set of multicast groups each consisting of exactly one sender and more than one receivers. $\alpha(s)$ is the multicast group where the sender is $n_s$. $\alpha(s)$ includes $n_s$. $e(s, \alpha(s))$ is the set of optimum routes from $n_s$ to each member of $(\alpha(s) - s)$.

Each node $n_i$ broadcasts HELLO messages at regular intervals. All nodes residing within the radio circle of $n_i$, reply with acknowledgment(ACK) message. Components of a HELLO messages are as follows:

(1) message type id(1) (2) sender id($n_i$), (3) sender location($x_i(t), y_i(t)$), (4) radio range ($rad(i)$), (5) maximum energy ($max\_eng(i)$), (6) current residual energy ($res\_eng(i)$), (7) rate of energy depletion ($depl\_eng(i)$), and (8) current time stamp ($t$).

Location of each node $n_i$ at time $t$ is denoted as an ordered pair $(x_i(t), y_i(t))$ where $x_i(t)$ and $y_i(t)$ indicate latitude and longitude, respectively, of the same node at time $t$. Radio range is the radius of the radio circle. Maximum energy is the highest possible battery power. $res\_eng(i)$ and $depl\_eng(i)$) are residual energy and energy depletion rate, respectively of $n_i$ at the current timestamp.

There exist some similarities between the attributes of HELLO and ACK messages. Attributes of an ACK message generated by a 1-hop downlink neighbor $n_j$ of $n_i$, are as follows:

(1) message type id(2), (2) sender id($n_i$), (3) sender location($x_i(t), y_i(t)$), (4) radio range ($rad(i)$), and (5) current time stamp ($t$).

Attributes of a RREQ message generated by a source node $\mathrm{n_s}$ for destination $\mathrm{n_d}$, are as follows:

(1) message type id(3), (2) source id($n_s$), (3) destination id($n_d$), (4) source location ($x_s(t), y_s(t)$), and (5) current time stamp ($t$).

Let $n_i$ be a downlink neighbor of $n_s$ that has received the RREQ message from $n_s$ at time $t'$. $n_i$ knows it's own residual battery power, maximum battery power, and energy depletion rate. Also, it knows values of the same attribute of $n_s$ too from the most recent HELLO message sent by $n_s$. Based on all these information, it can easily calculate expected residual lifetime of the link $n_s \rightarrow n_i$, denoted as $ERL(s, i)$. The calculation is performed using the procedure mentioned in Sect. 3. After that, $n_i$ will append three fields to the RREQ. They are: own id($n_i$), minlife (which is initially set to $ERL(s, i)$), and timestamp of initiation (or t) to the RREQ. Current timestamp information will be updated by $n_i$ to $\mathrm{t}^{\sim}$ overwriting the earlier current timestamp which was t.

Now assume that a node $n_j$ receives the RREQ from $n_i$ at time t, then $n_j$ will calculate $ERL(i, j)$. If $ERL(i, j)$ is less than $ERL(s, i)$, then new *minlife* will be $ERL(i, j)$; otherwise, *minlife* will remain set to $ERL(s, i)$. Please note that in the RREQ generated by the actual source of communication, no attribute like *minlife* and timestamp of initiation were present. But in the RREQs forwarded by the routers, there are attribute with those names. $n_j$ will change current timestamp to t whereas *timestamp* of initiation will remain t. Format of the three RREQs generated by $n_s$ and forwarded by $n_i$ and $n_j$ as below:

RREQ generated by $n_s$: $< 3, n_s, n_d, x_s(t), y_s(t), t >$
RREQ forwarded by $n_i$: $< 3, n_s, n_d, x_s(t), y_s(t), t', n_i, ERL(s, i), t >$
RREQ forwarded by $n_j$: $< 3, n_s, n_d, x_s(t), y_s(t), t', n_i, n_j, ERL(s, i), t >$

As soon as the RREQ packets arrive at the destination, the destination $n_d$ assigns weights to the route and selects the one with the highest weight. If weights of two routes are same, the one with smaller delay is preferred. If tie situation persists even after considering the delay, the route with lesser hop count is selected for communication. The number of hops can be easily computed from the RREQ packets where id numbers of all routers are present. If hop count of two competing routes having same the weight and delay are equal, then any one of them can be selected arbitrarily by the destination. After that destination node sends a route reply (RREP) packet to the source. Components of RREP are as follows:

(1) message type id(4), (2) sender id($n_d$), (3) source id($n_s$), (4) initiation time ($t$), (5) node id sequence in optimum route, and (6) current time stamp ($t$)

Items of a data packet is like:

(1) message type id(5), (2) source id($n_s$), (3) destination id($n_d$), (4) current time stamp ($t$), (5) data packet sequence number, and (6) actual data.

## 4 Optimum Route Selection

### 4.1 Estimating Route Life

Before optimum route selection, we need to compute lifetime of its links. Lifetime of a link from $n_i$ to $n_j$ at the current time is indicated as ERL(i, j) and defined in Eq. (1).

$$ERL(i,j) = Min(elife(i), elife(j), vlife(i,j)) \tag{1}$$

where *elife* is concerned with energy related link life; similarly, *vlife* is concerned with velocity related link life. *elife* is measured in Eq. (2) whereas *vlife* is measured in Eq. (5).

$$elife(i) = \frac{res\_eng(i) - max\_eng(i) \times 0.4}{depl\_eng(i)} \tag{2}$$

In order to compute *vlife*(*i*) consider last *v* number of ACK messages sent by a neighbor $n_j$ of $n_i$ to $n_i$. Let *p_trans*(*i*) denotes the maximum transmission power of $n_i$ and *disttl*(*i*, *j*) is the distance between $n_i$ and $n_j$ as per the power of the signal l-th ACK received by $n_j$ from $n_i$. The l-th ACK of $n_i$ is received by $n_j$ with signal power *p_recv*(*j*, *l*). Also assume that the time difference between two consecutive ACK messages is *tme*. As per Frii's transmission equation $distt_l(i,j$ is calculated by Eq. (3).

$$distt_l(i,j) = \sqrt[m]{\frac{p\_trans(i) \times K}{p\_recv(j,l)}} \tag{3}$$

For $2 \leq l \leq$v, if $distt_l(i,j) < distt_{l-1}(i,j)$, then Then the relative mobility *rmv*(*i*, *j*) between $n_i$ and $n_j$ is given by Eq. (4),

$$rmv(i,j) = \sum_{l=0}^{v} \frac{distt_l(i,j) - distt_{l-1}(i,j))}{tme \times v \times rad(i)} \tag{4}$$

where *rad*(*i*) is the radio range of $n_i$. If *rmv*(*i*, *j*) is less than 0.01, then *vlife* of $n_i \rightarrow n_j$ is assumed to be ∞. Otherwise, it is computed as in Eq. (5). Let the current distance between $n_i$ and $n_j$ be *dt*(*i*, *j*). Then minimum distance $n_j$ needs to cover to get out of the radio-range is $(rad(i) - dt(i,j))$. Therefore, *vlife* is formulated in Eq. (5)

$$vlife(i,j) = \frac{rad(i) - dt(i,j)}{rmv(i,j)} \tag{5}$$

Let R be a route s.t.

R: $n_s = ni \rightarrow n_{i+1} \rightarrow n_{i+2} \rightarrow\rightarrow n_{i+k} = n_d$

So

$$minlife(R) = \min(ERL(i, i+1), ERL(i+k-1, i+k)) \tag{6}$$

In a multipath routing protocol, multiple route options are available and hence, following different cases may occur.

#### 4.1.1 Case-1: At Least One Route Options Have Minlife Higher than the Completion Time of Live Sessions

The route with the highest weight is elected. In case of equal weight, the one producing least delay is preferred. If delay is also equal, we elect the route with minimum hop-count. Also, hop-counts may be equal. In that case, higher than the completion time of live sessions anyone may be chosen arbitrarily.

#### 4.1.2 Case-2: No Route Options have Minlife Higher than the Completion Time of Live Sessions

The one with highest *minlife* is chosen for data transfer. Equal *minlife* is handled as in case-1.

## 5 Computation of Weight

The weight of individual routes depends upon friends of all involved nodes as well as a maximum number of hops between consecutive nodes in a route having friends. These are mathematically expressed in the Definitions 1 and 2.

**Definition 1** If in a route R: $n_s = n_i \to n_{i+1} \to\to n_{i+k} \to\to n_{i+p} = n_d$, a router node $n_{i+\beta}$ as an alternative route to a successor$n_{i+\beta+\psi}$ such that $(0 < \beta < p), (2 \le \psi \le (p - \beta))$, no node in the alternative route is a part of R except $n_{i+\beta}$ and $n_{i+\beta+\psi}$, and *minlife* of an alternative route from ni+ to ni++ is greater than or equal to minlife of direct link from $n_{i+\beta}$ to $n_{i+\beta+\psi}$ in R, then $n_{i+\beta+\psi}$ will be called the friend of $n_{i+\beta}$. As far as source $n_s$ and destination $n_d$ are concerned, they are by default friends of themselves. Nodes having no friends are termed isolated. A node may have more than one friend and a node may be a friend of more than one node.

For example, consider the route R: $n_1 \to n_2 \to n_3 \to n_4$ in Fig. 4. $n_1$ (colored in red) is the source and $n_4$ (colored in blue is the destination. $n_1$ is the only friend of $n_1$ and $n_4$ is the friend of $n_4$ as per Definition 1. The direct route from $n_2$ to successor $n_4$ is $n_2 \to n_3 \to n_4$. There exists an alternative route from $n_2$ to $n_4$ i.e. $n_2 \to n_5 \to n_6 \to n_4$. $n_5$ and $n_6$ are not parts of R. If *minlife* of an alternative route from $n_2$ to $n_4$ is not less than the same of the direct route between the same pair of nodes, $n_4$ will be termed as a friend of $n_2$. $n_3$ has no friend, so it is isolated.

**Definition 2** *maxfriendgap* of a route R is the gap between two farthest consecutive non-isolated nodes in R, in terms of a number of hops.

For example, please consider route R: $n_1 \to n2 \to n_3 \to n_4$ in Fig. 1. $n_1, n_2$ and $n_4$ have friends. $n_3$ is isolated. Hop gap between $n_1$ and $n_2$ is 1; the same between $n_2$ and $n_4$ is 2. Therefore, $maxfriendgap(R) = 2$

**Lemma 1:** *Friend relation is reflexive only for source and destination nodes in a multicast session. For ordinary routers, it is non-reflexive.*

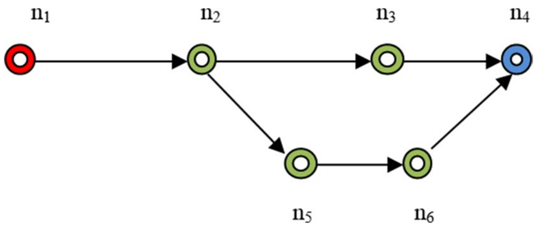

**Fig. 1** Finding friends

**Lemma 2:** *Friend relation is asymmetric.*

*Explanation* Links in ad hoc networks are not necessarily bi-directional. Therefore, if $n_i$ is a friend of $n_j$, then $n_j$ may not be a friend of $n_i$. For example, please consider Fig. 1. $n_4$ is a friend of $n_2$ but $n_2$ is not a friend of $n_4$.

**Lemma 3:** *Friend relation is transitive.*

Let in a route R, we have the followings:

(1) $n_{i+\beta+\psi} \in friend(n_{i+\beta})$
(2) $n_{i+\beta+\psi+v} \in friend(n_{i+\beta})$

Then, $n_{i+\beta}$ has an alternative route R′ to $n_{i+\beta+\psi}$, and $n_{i+\beta+\psi}$ has an alternative route R″ to $n_{i+\beta+\psi+v}$ such that,

(1) $R' : n_{i+\beta} = n_j \rightarrow n_{j+1} \rightarrow \cdots \rightarrow n_{j+g'} = n_{i+\beta+\psi}$
(2) $R'' : n_{i+\beta+\psi} = n_m \rightarrow n_{m+1} \rightarrow \cdots \rightarrow n_{m+g''} = n_{i+\beta+\psi+v}$
(3) $minlife(R') \geq minlife(R)$
(4) $minlife(R'') \geq minlife(R)$ A new route R‴ may be created as follows:

$$R''' : n_{i+\beta} = n_{i+\beta} = n_j \rightarrow n_{j+1} \rightarrow \cdots \rightarrow n_{m+g''} = n_m \rightarrow n_{m+1} \rightarrow \cdots \rightarrow n_{m+g''} = n_{i+\beta+\psi+v}$$
$$minlife(R''') = \min(minlife(R'), minlife(R''))$$

Hence, $minlife(R\ ) \geq minlife(R)$
So, $n_{i+\beta}$ has an alternative route to $n_{i+\beta+\psi+v}$, i.e. $n_{i+\beta+\psi+v}$ is a friend of $n_{i+\beta}$.
Weight W(R) of the route R is computed in Eq. (7).

$$W(R) = \frac{nonIso(R) \times avg(R) \times rcv(R)}{maxFriendGap(R)}; \quad noniso(R) = |nis(R)| \tag{7}$$

where $nis(R)$ is the set of non-isolated nodes in R.

$$avgf(R) = \frac{friend(v)}{totnod(R}; \quad n_v \in nis(R) \tag{8}$$

$friend(v)$ is the number of friends of node $n_v$ whereas $totnod(R)$ is total number of nodes in R. $rcv(R)$ is the number of multicast receivers belonging to route R or connected to R through or without routers. $maxFriendGap(R)$ is already illustrated earlier.

The above formulation is based on the observations below

(1) It is good for route R if most of the nodes have friends.
(2) A good number of friends on an average is an indication of strong alternative path strength.
(3) If more than one multicast receiver can be connected in a single path, then it will surely reduce message overhead in the system.

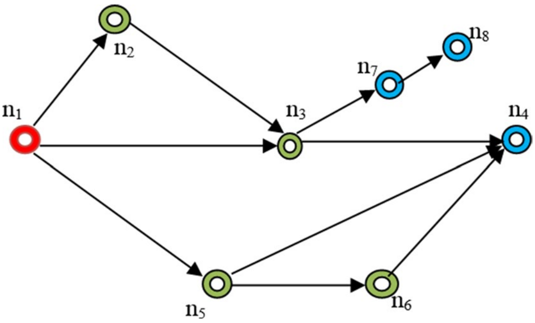

**Fig. 2** Routes from $n_1$ to $n_4$

**Table 1** Energy information about nodes

| Node-id | res-eng (mj) | max-eng (mj) | depl-eng (mj / ms) | Elife (ms) |
|---|---|---|---|---|
| n1 | 200 | 200 | 3 | 40 |
| n2 | 50 | 80 | 2 | 9 |
| n3 | 100 | 150 | 2 | 20 |
| n4 | 200 | 300 | 4 | 20 |
| n5 | 150 | 200 | 2 | 35 |
| n6 | 130 | 200 | 1 | 50 |
| n7 | 100 | 100 | 1 | 100 |
| n8 | 100 | 100 | 1 | 100 |

**Table 2** Position information about nodes

| Node-id | Coordinates | Radio-range |
|---|---|---|
| n1 | (0, 10) | 76 |
| n2 | (50, 50) | 45 |
| n3 | (70, 10) | 15 |
| n4 | (84, 10) | 10 |
| n5 | (80, 0) | 12 |
| n6 | (82, 0) | 11 |
| n7 | (75, 20) | 3 |
| n8 | (75, 21) | 4 |

### 5.1 Illustration of WTMR with Example

Consider Fig. 2, where $n_1$ multicast is source whereas $n_4$ and $n_7$ are multicast destinations. Maximum energy, residual energy, and energy depletion rates of these nodes appear in Table 1, whereas position related parameters appear in Table 2. All nodes except $n_2, n_7$ and $n_8$ are static. The velocities of $n_2, n_7$ and $n_8$ are 2m/ms, 1m/ms and 1m/ms. $n_2$ moves away from $n_1$ along the straight line connecting $n_1$ and $n_2$ as shown in the Fig. 2. Similarly, $n_7$ moves away from $n_3$ along the straight line connecting $n_3$ and $n_7$; $n_8$ moves away from $n_7$ along the straight line connecting $n_7$ and $n_8$.

### 5.1.1 For Multicast Receiver $n_4$

Node $n_2$ is connected to ony $n_1$ and $n_3$

$$\begin{aligned}
&\mathit{vlife}(1,2) = (7664.03)/2\,\text{ms i.e. } 5.985\,\text{ms}\\
&\mathit{vlife}(2,3) = (4544.72)/1\,\text{ms i.e. } 0.28\,\text{ms}\\
&\mathit{So}, \mathit{ERL}(1,2) = \mathit{min}(40,9,5.985) = 5.985\\
&\mathit{ERL}(2,3) = \mathit{min}(9,20,0.28) = 0.28\\
&\mathit{ERL}(3,4) = \mathit{min}(20,20,\infty) = 20\\
&\mathit{ERL}(1,3) = \mathit{min}(40,20,\infty) = 20
\end{aligned}$$

Similarly,

$$\begin{aligned}
&\mathit{ERL}(1,5) = \mathit{min}(40,35,\infty) = 35\\
&\mathit{ERL}(5,6) = \mathit{min}(35,50,\infty) = 35\\
&\mathit{ERL}(6,4) = \mathit{min}(50,20,\infty) = 20\\
&\mathit{ERL}(5,4) = \mathit{min}(35,20,\infty) = 20
\end{aligned}$$

Route options from $n_1$ to $n_4$ are:

$$\begin{aligned}
&\text{R1: } n_1 \xrightarrow{(5.985)} n_2 \xrightarrow{(0.28)} n_3 \xrightarrow{(20)} n_4\ (\text{minlife} = 0.28)\\
&\text{R2: } n_1 \xrightarrow{(20)} n_3 \xrightarrow{(20)} n_4\ (\text{minlife} = 20)\\
&\text{R3: } n_1 \xrightarrow{(35)} n_5 \xrightarrow{(35)} n_6 \xrightarrow{(20)} n_4\ (\text{minlife} = 20)\\
&\text{R4:} n_1 \xrightarrow{(35)} n_5 \xrightarrow{(20)} n_4\ (\text{minlife} = 20)
\end{aligned}$$

Let $n_1$ generates packets after every 3 ms and overall 5 packets are to be transferred from source to destination. Also assume that the time required by R1, R2, R3, and R4 to transfer one data packet to the destination, is 5ms, 1ms, 2ms and 1ms respectively. Hence, overall delays (*ov_del*) of R1, R2, R3 and R4 are calculated as follows:

$$\begin{aligned}
&\mathit{ov\_del}(R1) = 5 + (5-1)\times 3 = 17\,\text{ms}\\
&\mathit{ov\_del}(R2) = 1 + (5-1)\times 3 = 13\,\text{ms}\\
&\mathit{ov\_del}(R3) = 2 + (5-1)\times 3 = 14\,\text{ms}\\
&\mathit{ov\_del}(R3) = 1 + (5-1)\times 3 = 13\,\text{ms}
\end{aligned}$$

R1 will most probably break because its *minlife* is less than its *ov_del*. Among the options R2, R3 and R4, all are expected to survive till end of the session; *minlife* of all are 20, higher than their respective *ov_del*. The weight of these three paths are computed as follows:

$$\begin{aligned}
W(R2) &= \frac{(2\times 2\times 1)}{(3\times 2)} = 0.6667\\
W(R3) &= \frac{(3\times 3\times 1)}{(4\times 2)} = 1.125\\
W(R4) &= \frac{(3\times 3\times 1)}{(3\times 1)} = 3
\end{aligned}$$

Please note that $n_1$ is a friend of $n_4$ and $n_4$ is a friend of $n_1$, because they are source and destinations, respectively. $n_3$ in R2 does not have a friend. So, *maxFriendGap* of R2 is 2, *noniso* is 2 and *avgf* is (2/3). In R3, $n_5$ has a friend $n_4$. $n_6$ doesn't have any friend. Therefore, *maxFriendGap* of R3 is also 2. Its *noniso* is 3 and *avgf* is (3/4). In R4, $n_1$, $n_5$ and $n_4$ all three have friends. Therefore, *maxFriendGap* is 1. Its *noniso* is 3 and *avgf* is (3/3) i.e. 1. All these route options connect only one multicast receiver. Among the options R2, R3 and R4, R4 is selected for communication since $W(R4)$ is highest.

### 5.1.2 For Multicast Receiver $n_7$

Possible route options fro $n_1$ to $n_7$ are as follows:

R1: $n_1 \rightarrow n_2 \rightarrow n_3 \rightarrow n_7$
R2: $n_1 \rightarrow n_3 \rightarrow n_7$

As already mentioned,

$$ERL(1,2) = min(40, 9, 5.985) = 5.985$$
$$ERL(2,3) = min(9, 20, 0.28) = 0.28$$
$$ERL(1,3) = min(40, 20, \infty) = 20$$
$$vlife(3,7) = \frac{(15-11.18)}{1} = 3.82\,\text{ms}$$
$$ERL(3,7) = min(20, 20, 3.82) = 3.82$$

Let $n_1$ generates packets after every 0.2 ms and overall 2 packets are to be transferred from source to destination. Assume that time required by R1 and R2 to transfer one data packet to destination, is 0.05 ms and 0.02 ms respectively. Then overall delays $ov\_del(R1)$ and $ov\_del(R2)$ are calculated as follows:

$$ov\_del(R1) = 0.05 + (2-1) \times 0.2 = 0.25\,\text{ms}$$
$$ov\_del(R2) = 0.2 + (2-1) \times 0.2 = 0.22\,\text{ms}$$
$$minlife(R1) = min(5.985, 0.28, 3.82) = 0.28$$
$$minlife(R2) = min(20, 3.82) = 3.82$$

Please note that, $minlife(R1) > ov\_del(R1)$ and $minlife(R2) > ov\_del(R2)$. So, both routes are expected to remain alive till the multicast session is over.

For R1:

$$noniso = 2, \quad avgf = \frac{(1+0+0+1)}{4} = 0.5, \quad rcv = 2, \quad maxFriendGap(R1) = 3$$

So, $W(R1) = \frac{(2\times0.5\times2)}{3} = 0.6667$

For R2:

$$noniso = 2, \quad avgf = \frac{(1+0+1)}{3} = 0.6667, \quad rcv = 2, \quad maxFriendGap(R2) = 2$$

So, $W(R2) = \frac{(2\times0.6667\times2)}{2} = 1.3333$

Clearly, weight of R2 is high, therefore R2 is used for communication from $n_1$ to $n_7$.

### 5.1.3 For Multicast Receiver $n_8$

$n_7$ and $n_8$ move with the same velocity in the same direction. Therefore *vlife*(7, 8) is $\infty$. All routes from $n_1$ to $n_8$ pass through $n_7$. So, optimum route from $n_1$ to $n_7$ is the optimum route from $n_1$ to $n_8$.

## 6 Simulation Results

### 6.1 Simulation Environment

Performance of the proposed multicast algorithm WTMR is studied with respect to MAODV, ODMRP and EEMR in NS-2 network simulator. Metrics are the packet delivery ratio, control message overhead, multicast route lifetime, and end-to-end delay. All are measured with respect to a number of nodes, node speed and number of senders. MAODV and ODMRP are state-of-the-art representatives of tree-based and mesh-based multicast protocols respectively. EEMR particularly focuses on energy efficient perspective.

In all the experiments, ad hoc network is modeled randomly. A number of nodes is 20, 40, 60, 80 and 100. Simulation area has size 1000 m × 1000 m. Mobility model is random waypoint where traffic type is constant bit rate. Velocity of nodes can take different value 10 km/h, 20 km/h, 30 km/h, 40 km/h and 50 km/h. A number of simultaneous senders range from 5 to 20 while group size also ranges from 5 to 20. Used MAC protocol is IEEE 802.11 g. Broadcast channel capacity is 2 Mbps. Traffic sources generate traffic at a rate 20 packets/s. Size of each packet is 512 bytes. The nodes are equipped with queues for storing packets before forwarding. The maximum size of the queue is 100. Radio-range varies from 50 to 300 m.

### 6.2 Experimental Results

Performance comparisons corresponding to different metrics appear in the Fig. 3 to Fig. 14.

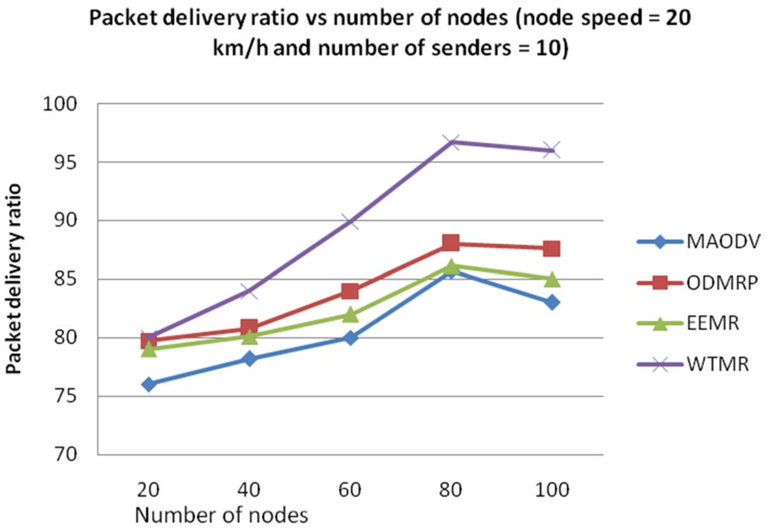

**Fig. 3** Packet delivery ratio versus number of nodes

### 6.2.1 Packet Delivery Ratio

Packet delivery ratio is the percentage of data packets successfully received by multicast destinations with respect to the number of data packets transmitted to them. It implies efficiency of the routing technique. Figure 3 shows the packet delivery ratio w.r.t. the number of nodes. Compared to ODMRP, MAODV and EEMR, WTMR produces much better packet delivery ratio. The reason is that none of the protocols ODMRP, MAODV and EEMR consider route lifetime. EEMR is energy efficient but high energy efficiency does not always guarantee high residual lifetime. Energy depletion rates need to be considered too. Also, WTMR gives priority to the routes that are comparatively stable (i.e. are capable of surviving throughout the multicast session, and equipped with alternative paths) and can connect more than one multicast receivers in a single route. Choosing such routes might require one stable route to deliver multicast messages from source to all multicast destinations. Similarly, if we measure with respect to a number of senders, multicast forwarding loads on nodes increase for all the protocols. As a result, nodes with high residual lifetime and alternative paths are expected to survive while the others are supposed to deplete easily. Therefore, WTMR delivers more packets successfully to destinations than its competitors as shown in Fig. 7. Figure 5 graphically investigates the packet delivery ratio with respect to node speed. As node speed increases, new links start to appear while old links break frequently. This is the situation where link lifetime consideration comes up with lots of benefits. Routes that last long or have alternatives are expected to produce, less control overhead, less contention and collision and therefore, better packet delivery ratio.

In Figs. 4 and 5, packet delivery ratio decreases for all the protocols with an increase in node speed and number of senders. On the other hand, in Fig. 3 initially, packet delivery ratio an increases due to more link formation with an increase in the number of nodes. After that, it starts decreasing as the number of nodes arrives at saturation.

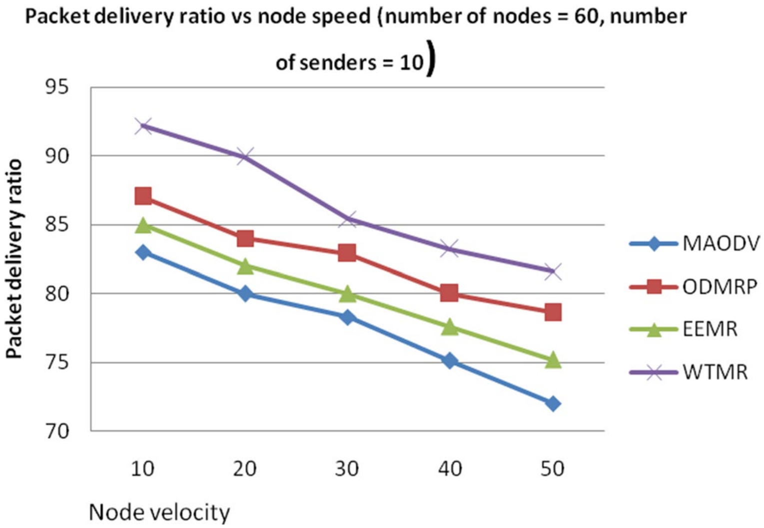

**Fig. 4** Packet delivery ratio versus node speed

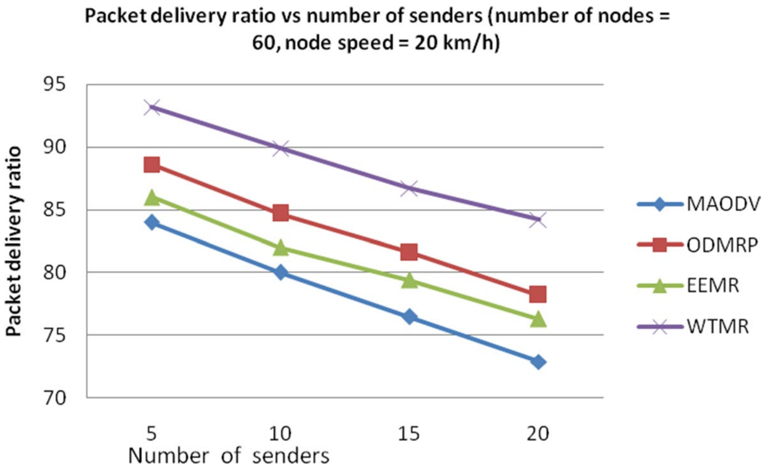

**Fig. 5** Packet delivery ratio versus number of senders

### 6.2.2 Control Message Overhead

Control overhead is the summation of extra control messages transmitted per second in the network for repairing routes, during simulation. As one particular route breaks, in order to repair that, route-requests (RREQs) packets have to be flooded throughout the network. WTMR gives higher weight to the routes having alternatives so that if a link to the original route breaks, it's alternative can be tried. With an increase in the number of nodes, so many such alternatives appear feasible, which, if used, eliminate the need of injecting RREQ packets within the network. This greatly reduces control overhead in WTMR compared to ODMRP, MAODV and EEMR. ODMRP is mesh-based. So, its control overhead is less than MAODV and EEMR. Please note that, MAODV uses only hop count for route selection. Figures 6, 7 and 8 show control overhead with respect to

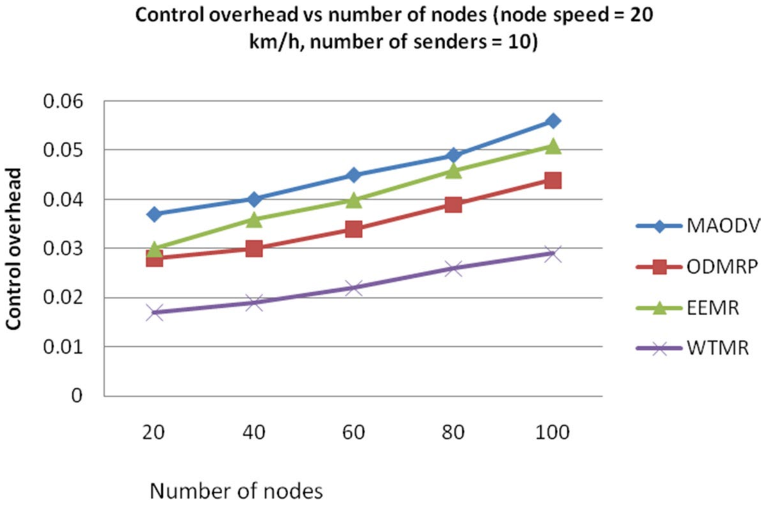

**Fig. 6** Control message overhead versus number of nodes

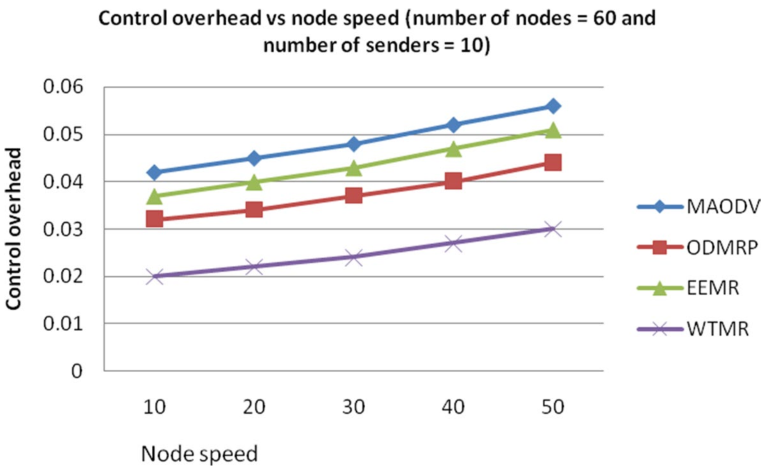

**Fig. 7** Control message overhead versus node speed

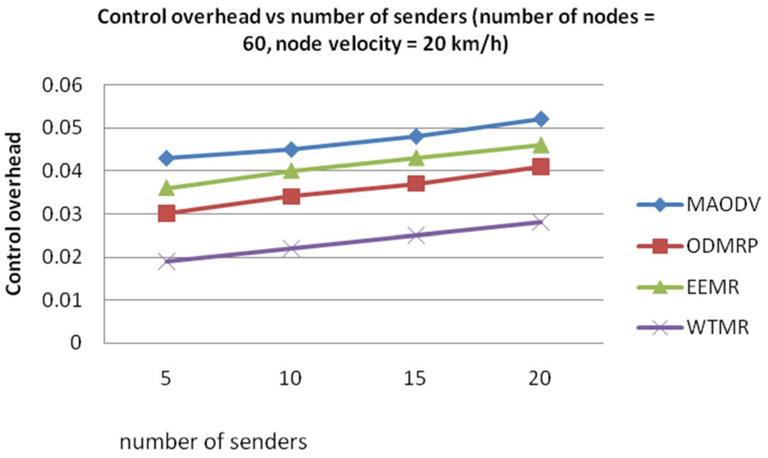

**Fig. 8** Control message overhead versus number of senders

a number of nodes, node velocity, and number of senders, respectively. As expected, control overhead increases with an increase in node velocity and the number of senders.

### 6.2.3 Multicast Route Lifetime

Multicast route lifetime is defined by the average time period before first route-repair in any multicast session after first data packet is transmitted by the sender. If no route-repair is required then time period between transmission of first data packet and delivering last data packet will be a lifetime of that multicast route. Unlike the competitors, WTMR directly considers route lifetime which is much different from energy efficiency in EEMR. Support of alternative routes strengthen the lifetime efficiency of WTMR.

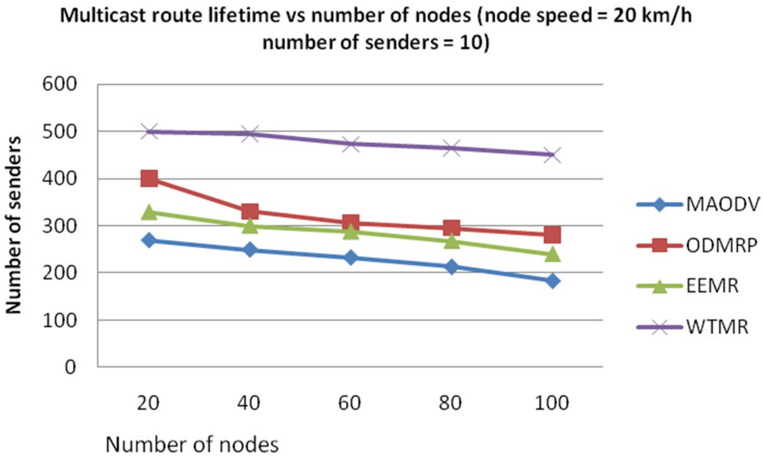

**Fig. 9** Multicast route lifetime versus number of nodes

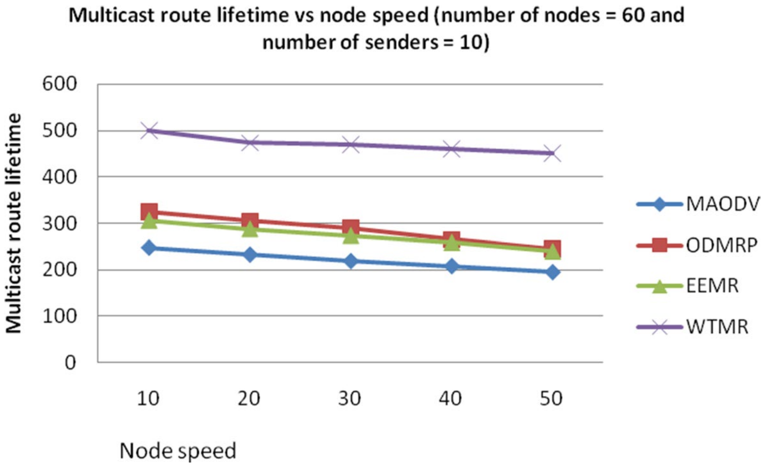

**Fig. 10** Multicast route lifetime versus node speed

The effects can be seen in Figs. 9, 10 and 11. As the number of nodes or number of senders increases, nodes deplete more energy per second and their lifetimes start reducing. Lifetime in WTMR has two components energy related lifetime and velocity related lifetime. The relative velocity of nodes is considered in order to ensure stability.

### 6.2.4 End-To-End Delay

End-to-end delay is defined as the time elapsed between transmission of RREQ by the multicast sender at the beginning of a multicast session and time of receiving last data packet by the last destination. WTMR saves the time of repairing routes by reducing a

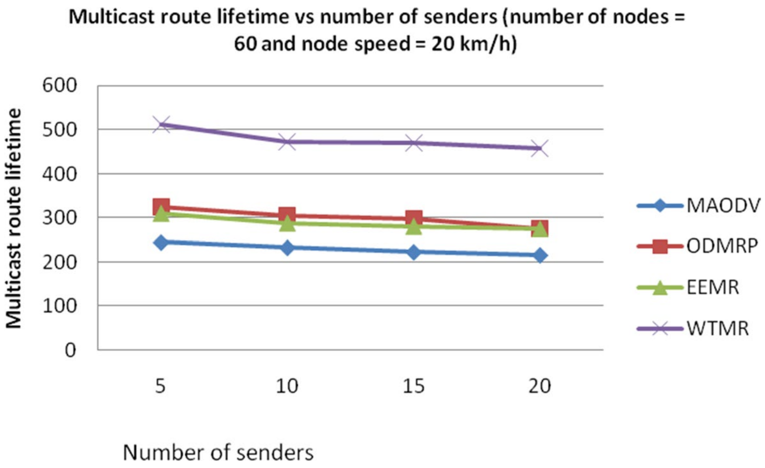

**Fig. 11** Multicast route lifetime versus number of senders

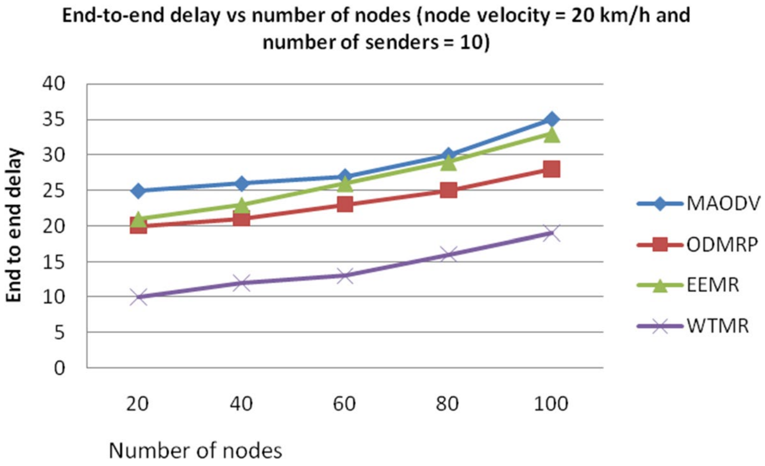

**Fig. 12** End-to-end delay versus number of nodes

number of route discovery sessions since the protocol is lifetime efficient. Moreover, due to a reduction in control overhead in WTMR, message contention and collision reduces, saving the time required for route rediscovery and resending of data packets. Improvements in favor of WTMR can be seen in the Figs. 12, 13 and 14.

## 7 Conclusion

The main strength of this WTMR is consideration of expected residual lifetime by energy depletion rate and alternative paths. WTMR is an independent of the underlying unicast routing protocol which mainly give importance to two parameters of ad hoc

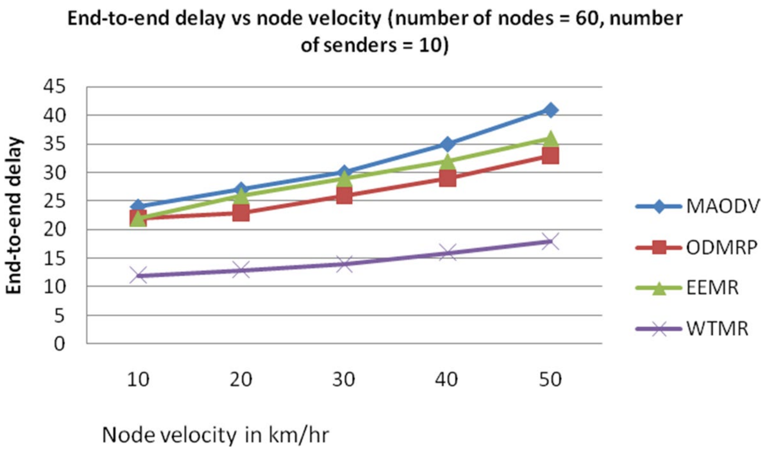

**Fig. 13** End-to-end delay versus node speed

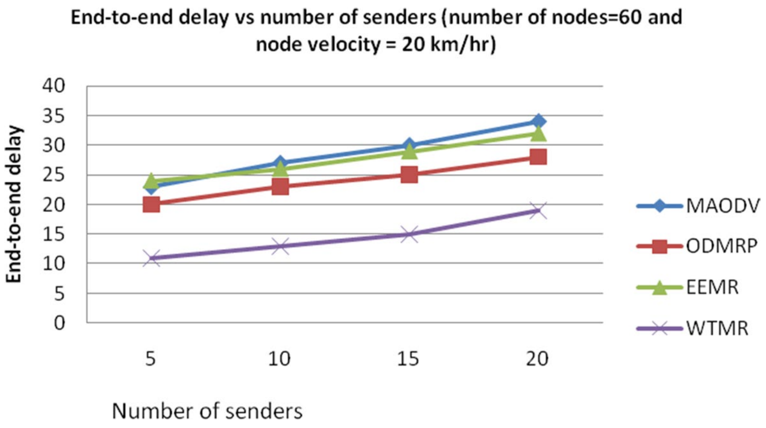

**Fig. 14** End-to-end delay versus number of senders

networks viz veleocity of node and depletion of residual energy. Assigning priority to the nodes having high residual lifetime and more friends with low mobility, greatly reduces the number of link breakages, control overhead, and delay.

## References

1. Chlamtac, I., Conti, M., & Liu, J. (2003). Mobile ad hoc networking: Imperatives and challenges. *Ad Hoc Networks*, *1*, 13–64.
2. Corson, S., & Macker, J. (1999). Mobile ad hoc networking (manet): Routing protocol performance issues and evaluation considerations. https://tools.ietf.org/html/rfc2501.html.
3. L., S.-J., et al. (2000). A performance comparison study of ad hoc wireless multicast protocols. *Proceeding of INFOCOM*, *2000*, 564–574.

4. de Morais Cordeiro, C., Gossain, H., & Agrawal, D. (2003). Multicast over wireless mobile ad hoc networks: Present and future directions. *IEEE Network*, *17*, 52–59.
5. Obraczka, K., & Tsudik, G. (1998). Multicast routing issues in ad hoc networks. In *Proceeding of IEEE international conference on universal persona communication (ICUPC98).*
6. Soni, S. K. & Aseri, T. C. (2010). A review of current multicast routing protocol of mobile ad hoc network. In *Proceeding of second international conference on computer modeling and simulation ICCMS' 10* (Vol. 3, pp. 207–211).
7. Banerjee, A., Dutta, P., & Ghosh, S. (2012). Fuzzy-controlled energy-efficient weight-based two hop clustering for multicast communication in mobile ad hoc networks. *Proceeding of the SEMCCO*, *2011*, 520–529.
8. Sajama, S., & Haas, Z. J. (2001). Independent-tree ad hoc multicast routing (itamar) (pp. 7803–7805). https://people.ece.cornell.edu/haas/Papers/VTC01-sajama-haas.pdf.
9. Wang, B. & Gupta, S. K. S. (2003). S-remit: A distributed algorithm for source-based energy efficient multicasting in wireless ad hoc networks. In *Proceeding of GLOBECOM 2003* (Vol. 6, pp. 3519–3524).
10. Zhong, M., Fu, V., & Jia, X. (2011). Maodv multicast routing protocol based on node mobility prediction. In *Proceeding of international conference on E-business and E-government (ICEE).*
11. Lee, S.-J., Gerla, M., & Chiang, C.-C. (1999). On-demand multicast routing protocol. In *Proceeding of IEEE WCNC99* (pp. 1298–1304).
12. Xie, J., Talpade, R. R., McAuley, A., & Liu, M. (2002). Amroute: Ad hoc multicast routing protocol. In *Mobile networks and applications* (Vol. 7, pp. 429–439). Kluwer Academic Publishers.
13. Yadav, A. K., & Tripathi, S. (2015). Dlbmrp: Design of load balanced multicast routing protocol for wireless mobile ad-hoc network. *Wireless Personal Communications*, *85*, 1815–1829.
14. Olagbegi, B. S., & Meghanathanl, N. (2010). A review of the energy efficient and secure multicast routing protocols for mobile ad hoc net works. *International Journal on Applications of Graph Theory in Wireless Ad Hoc Net Works and Sensor Networks (GRAPH-HOC)*, *2*, 1–15.
15. Tavli, B., & Heinzelmanl, W. B. (2011). Energy-efficient real-time multicast routing in mobile ad hoc net works. *IEEE Transactions on Computers*, *60*, 707–722.
16. Banerjee, A., Sufian, A., & Dutta, P. (2017). EMR-PL: Energy-efficient multipath routing based on link life prediction in ad hoc networks. *Journal of Information and Optimization Sciences*, *39*, 367–382.
17. Tiwari, V. K., & Malviyal, A. K. (2016). An energy-efficient multicast routing (eemr) protocol in MANET. *International Journal of Engineering and Computer Science*, *5*, 19139–19148.
18. Dutta, P., & Banerje, A. (2012). Fuzzy-controlled power-aware multicast routing (FPMR) for mobile ad hoc networks. In *Proceeding of the second international conference on computer, communication, control and information technology (C3IT 2012), procedia technology* (Vol. 4, pp. 38–49).
19. Wijesiri, G. P., Chowdhury, S. S., & Li, F. Y. (2016). Energy harvesting-aware backoff algorithms for distributed device-to-device communication. In *Proceeding of 2016 IEEE 83rd vehicular technology conference (VTC spring).*
20. Singh, S., Woo, M., & Raghavendra, C. S. (1998). Power-aware routing in mobile ad hoc networks. In *Proceeding of the 4th annual ACM/IEEE international conference on Mobile computing and networking (MobiCom '98)* (pp. 181–190).
21. Kamboj, P., & Sharma, A. K. (2010). Scalable energy efficient location aware multicast protocol for manet (seelamp). *Journal of Computing*, *2*, 20–30.
22. Roy, A., Hazarika, M., & Debbarma, M. K. (2012). Energy efficient cluster based routing in manet. In *Proceeding of international conference on communication, information and computing technology (ICCICT).*
23. Banerjee, A., Dutta, P., & Sufian, A. (2018). Fuzzy-controlled energy-efficient single hop clustering scheme with (FESC) in ad hoc networks. *International Journal of Information Technology*, *10*, 313–327.
24. Shenbagaraj, S., & Ramar, D . K. (2012). An optimized and energy efficient multicast routing based on genetic algorithm for wireless mobile ad hoc networks. *International Journal of Research in IT and Management*, *2*, 390–404.
25. Correia, L. M., Zeller, D., Blume, O., Ferling, D., Jading, Y., Gdor, I., et al. (2010). Challenges and enabling technologies for energy aware mobile radio networks. *IEEE Communications Magazine*, *48*, 66–72.

**Abu Sufian** received his B.Sc. (Hons) in Mathematics from University of Kalyani in 2007 and MCA from Kalyani Govt Engineering College under West Bengal University of Technology in 2010. Currently he is an assistant professor and the Co-ordinator of the Department of Computer Science, University of Gour Banga. He is currently pursuing his Ph.D. in Computer Science. He has published more than 13 research papers in several reputed journals and conferences both international and national including publishers namely IEEE, Springer etc. He is a co-inventor of one Indian Patents. His research areas including Adhoc Networks, Computational Intelligence, Deep Learning, Computer Vision. He is a member of IEEE and IEEE CIS, and also a life member of Computer Scoiety of India and other national and international societies such as The Computer Vision Foundation.

**Dr. Anuradha Banerjee** recived her B.E. in C.S.E from Bengal Engineering and Science University, Shibpur and Ph.D. from West Bengal University of Technology in 2001 and 2013 respectively. Currently she is an assistant professor of the Department of Computer Applications, Kalyani Govt Engg. College. She has published about 50 research papers in reputed journals and conferences both international and national. She has also written several book chapters in some international publications. She is a co-inventor of four Indian Patents(filed). She is a member of ACM and some other national and international societies. Her research interest including Mobile Ad-hoc Networks, Software Defined Networks, Fuzzy Theory etc.

**Dr. Paramartha Dutta** received his B.Stats, M.Stats and M.Tech. (C.S.E) from the Indian Statistical Institute, Kolkata in the years 1988, 1990 and 1993 respectively. He after wards completed his Ph.D. in Engineering from the Bengal Engineering and Science University, Shibpur. Currently he is a professor of the Department of Computer and System Sciences, Visva-Bharati University, India. He has co-authored eight books and has also seven edited books with leading publishing house such as Springer, Elsevier IGI Global etc. to his credit. He has published more than two hundred papers in various journals and conference proceedings, both international and national as well as several book chapters in edited volumes of reputed International publishing house like Elsevier, Springer-Verlag, CRC Press, John Wiley, IGI Global to name a few. He has guided six scholars who already had been awarded their Ph.D. Presently, he is supervising six scholars for their Ph.D. program. He is the co-inventor of six published Indian Patents and one filed. He is also a senior member of IEEE and IEEE CIS, Life member of Computer Society of India and many other reputed organizations.