Alternative Node Based Energy Depletion and Expected Residual Lifetime Balancing Method for Mobile Ad Hoc Networks

Anuradha Banerjee
Department of Computer Applications, Kalyani Govt. Engg College, West Bengal
Email: anuradha79bn@gmail.com

Paramartha Dutta
Department of Computer and Systems Science, Visva-Bharati University, Santiniketan, West Bengal
Email: paramartha.dutta@gmail.com

ABSTRACT

A mobile ad hoc network is an infrastructure less network, where nodes are free to move independently in any direction. The nodes have limited battery power; hence we require efficient balancing techniques (energy depletion or expected residual lifetime, whichever is applicable under specific circumstances) to reduce overload on the nodes, wherever possible, to enhance their lifetime and network performance. This kind of balance among network nodes increase the average lifetime of nodes and reduce the phenomenon of network partitioning due to excessive exhaustion of nodes. In this paper, we propose an alternative-node based balancing method (ANB) that channels the forwarding load of a node to some other less exhausted alternative node provided that alternative node is capable of handling the extra load. This greatly reduces the number of link breakages and also the number of route-requests flooded in the network to repair the broken links. This, in turn, improves the data packet delivery ratio of the underlying routing protocol as well as average node lifetime.

Keywords - Ad hoc network, Alternative Node, Energy Depletion, Link Breakage, Expected Residual Lifetime Balancing.

I. INTRODUCTION

An ad hoc network is a group of wireless mobile devices or nodes that communicate with each other in a collaborative way over multi-hop wireless links without any stationary infrastructure or centralized management. These networks are deployed mainly in battlefields and disaster situations such as earthquake, floods etc. Many routing protocols have been proposed for ad hoc networks. They can be mainly categorized as proactive and reactive routing protocols. Among proactive routing protocols, destination-sequence distance vector (DSDV) [1], wireless routing protocol (WRP) [2], global state routing (GSR) [3] and cluster-based gateway switch routing (CGSR) [4] are well known. In all proactive routing protocols the nodes proactively store route information to every other node in the network. In general, the proactive routing protocols suffer from extremely huge storage overhead because they store information both about active and non-active routes. This inculcates the unnecessary complexity of discovering routes to the destinations with which a node rarely communicates. Reactive or on-demand routing protocols are designed to reduce this overhead. In reactive routing protocols, when a source node needs to communicate with a destination, it floods route-request packets through out the network to discover a suitable route to the destination. Dynamic source routing (DSR) [5], ad hoc on-demand distance vector routing (AODV) [7], adaptive communication aware routing (ACR) [8], flow-oriented routing protocol (FORP) [9] and associativity-based routing (ABR) [10] are well-known among the reactive-based routing protocols. AODV builds routes using a route-request, route-reply query cycle. When a source node desires to send packets to a destination for which it does not already have a route, it broadcasts a route-request (RREQ) packet across the network. Nodes receiving this packet update their information for the source node and set up pointers backward to the source node in their routing tables. A node receiving the route-request (RREQ) packet sends a route-reply (RREP) if it is either the destination or has a recently established route to the destination with. Dynamic source routing (DSR) is similar to AODV in that it forms a route on-demand when a source node requests one. It uses source routing instead of relaying on the routing table at each device. Determining source routes require accumulating the address of each router in the route-request message. In FAIR [11], the source node transmits RREQ packets that arrive at the destination through multiple paths. Depending upon the locations, residual energy, velocity etc. various characteristics of the routers, the destination node evaluates performance of the paths by considering their stability and agility. Then communication from source to destination begins through one of the best paths. FORP and ABR are link stability based routing protocols that also rely on the flooding of...
RREQ packets for route discovery. So, if the number of RREQ packets can be reduced then much lesser number of routers will be involved in the route discovery process in the ANB versions of the above-mentioned routing protocols compared to their ordinary versions. As a result, network throughput or data packet delivery ratio enhances with decrease in energy consumption in nodes.

Our present article proposes an alternative-node based balancing method (ANB) technique for in ad hoc networks where the forwarding load of a node is transferred partly to an alternative one in a very specific manner so that the average longevity of the nodes increase. This reduces the phenomena like link breakage and network partitioning. Automatically it reduces the cost of message and average energy consumption in network nodes. This, in turn, decreases the number of packet collision in the network and improves the network throughput or data packet delivery ratio. Our proposed technique can be applied with any reactive routing protocol to enhance the performance of the protocol.

II. THE SCHEME OF ANB

The entire concept of ANB is dependent upon the notion of alternative nodes. It is defined below.

Definition: Alternative Node

A node $n_j$ is termed as an alternative to another node $n_i$ where $n_i \neq n_j$ (i.e. a node cannot be the alternative of its own), if $n_j$ has the same set of uplink and downlink neighbours of $n_i$ at the same time, except the nodes themselves. For example, let $U_i(t)$ and $D_i(t)$ denote the set of uplink and downlink neighbours of $n_i$ at time $t$. So, $n_i$ and $n_j$ will be alternatives provided the following conditions are satisfied:

i) $U_i(t) - \{ n_j \} = U_j(t) - \{ n_i \}$

ii) $D_i(t) - \{ n_j \} = D_j(t) - \{ n_i \}$

Please note that the alternative nodes need not necessarily be the neighbours (uplink or downlink) of one another.

2.1 How to find out alternatives

In ANB, each node transmits HELLO message within its radio-range at regular intervals which is received by its downlink neighbours. In response, the downlink neighbours transmit the ACK or acknowledgement message to sender of the HELLO message. The components of HELLO message transmitted by a node $n_i$ at time $t$ are given by,

i) Node identification number $n_i$

ii) Current timestamp $t$

iii) Identification number of uplink neighbours i.e. $U_i(t)$

iv) Identification number of downlink neighbours i.e. $D_i(t)$

v) Radio range rad(i)

vi) Geographical location i.e. $(x_i(t), y_i(t))$ where $x_i(t)$ is the latitude of $n_i$ at time $t$ and $y_i(t)$ is the longitude of the same node at the same time

vii) Forwarding load $p_i$ so far, in terms of number of packets forwarded per second (for these packets $n_i$ is not the source)

viii) Amount of energy $\eta_i$ required to forward each packet

ix) Total battery power $E_i$

x) Residual battery power $P_i$ at the current time

xi) Time $t_i$ of starting operation in the network

xii) Number of packets $p_i(t)$ transmitted as source so far by $n_i$

xiii) Total size $M_j$ of message queue

xiv) Number of messages $W_j$ waiting in the message queue at that time

On the other hands, the components of ACK message transmitted by a downlink neighbour $n_p$ of $n_i$ at time $t$, are given by

i) Sender identification number $n_p$

ii) Receiver identification number $n_i$

iii) Identification numbers of the alternatives of $n_i$

iv) Current timestamp $t$

As soon as a node $n_p$ receives two HELLO messages from two nodes $n_i$ and $n_j$ then $n_p$ compares between the uplink and downlink neighbour sets mentioned in those HELLO messages. If they satisfy the criteria of alternative nodes, then $n_p$ embeds the identification number $n_i$ in the ACK that it sends to $n_j$ and the identification number $n_j$ in the ACK that it sends to $n_i$. More than one such alternative can be found in such way. Among all the downlink neighbours of the alternative nodes, the one with least identification number performs all the balancing computations and informs the alternative nodes about their actual forwarding load for balanced environment for each alternative node in a message ALT. The components of ALT sent by $n_p$ are as follows:

i) Sender identification number $n_p$

ii) Receiver identification number $n_i$

iii) (Identification number of alternative $n_j$, balancing method , balanced forwarding load of $n_i$, balanced forwarding load $n_j$, $M_j$, $W_j$) for each alternative $n_j$ of $n_i$

The balancing method is both residual lifetime balancing and energy depletion balancing.

Note: The alternative relation is non-reflexive, symmetric and transitive. Its proof is given in the appendix section.

2.2 Balance of Energy Depletion and Residual Lifetime

At any point of time, let $n_i$ be an alternative of $n_i$. Balancing (energy depletion or residual lifetime) is performed as per the following cases. Without any loss of
generally these conditions are based on the assumption that \( n_i \) started operating after \( n_j \), i.e. \( t_i < t_j \):

**Case -1**

When \( n_i \) started operation, the residual energy of \( n_i \) at that time was higher than the total battery power of \( n_j \) and at present the residual battery power of \( n_i \) is less than the present residual battery power of \( n_j \), although the number of packets transmitted by \( n_i \) as source is lesser than the number of packets transmitted by \( n_j \) as source. These are mathematically expressed as follows:

i) \( (E_i - p_i \alpha_i (t_i - t_0) - pt_i \alpha_i) > E_j \)

ii) \( R_i < R_j \)

iii) \( p_i < p_j \)

The situation indicates that \( n_i \) has forwarded much more packets than \( n_j \). In this case, we go for residual energy balancing provided the average node lifetime doesn’t deteriorate (this is discussed in detail in section 2.3).

**Case -2**

If the conditions mentioned in case 1 are not true or somehow residual lifetime cannot be balanced then we try for energy depletion balancing.

### 2.3 Pair-wise Balance of Residual Lifetime

After balance of expected residual lifetime some of the forwarding load of \( n_i \) is channelled through \( n_j \) because at present the forwarding load \( p_i \) of \( n_i \) is greater than the forwarding load \( p_j \) of \( n_j \). For this channeling, \( n_i \) need to communicate with some of its uplink neighbours and need to inform them that they will forward the packets to \( n_j \) now instead of \( n_i \). Let, the upper limit of time duration for this kind of communication from \( n_i \) with uplink neighbours is given by \( \tau_{\text{max}} \). Also assume that \( R_i' \) and \( R_j' \) denote the residual energy of \( n_i \) and \( n_j \) after time duration \( \tau_{\text{max}} \) where \( R_i \) and \( R_j \) are initial energy of the nodes \( n_i \) and \( n_j \).

Then,

\[
R_i' = R_i - p_i \alpha_i \tau_{\text{max}}
\]

\[
R_j' = R_j - p_j \alpha_j \tau_{\text{max}}
\]

Let \( p_i' \) and \( p_j' \) denote forwarding load of \( n_i \) and \( n_j \) after residual lifetime balancing.

So, \((p_i' + p_j') = (p_i + p_j) \) (3)

After lifetime balancing, the new energy depletion rate of \( n_i \) is \( p_i' \) \( \alpha_i \) per second. Then the expected remaining lifetime of \( n_i \) after lifetime balancing is \((R_i' / (p_i' \alpha_i))\). The same of \( n_j \) is \((R_j' / (p_j' \alpha_j))\). These two are equal since residual lifetime has been balanced. So,

\[
R_i' / (p_i' \alpha_i) = R_j' / (p_j' \alpha_j)
\]

i.e. \( p_i' = p_j' (R_i' \alpha_i) / (R_j' \alpha_j) \)

Replacing \( p_j' \) by \([p_j' (R_i' \alpha_i) / (R_j' \alpha_j)]\) in (3) we get,

\[
p_j' (1 + (R_i' \alpha_i) / (R_j' \alpha_j)) = (p_j + p_j') \]

So, \( p_j' = (p_j + p_j') (R_j' \alpha_j) / ((R_j' \alpha_j + R_i' \alpha_i)) \)

If \( p_j' \) is a fraction, we take, \( p_j' = \lceil p_j' \rceil \). Putting this in (3) we get,

\[
p_i' = (p_i + p_j - p_j')
\]

\[
(8)
\]

### 2.4 Effect of Balancing Residual Lifetime on Average Node Lifetime

Initial residual lifetime \( IL_i \) of node \( n_i \) and the same \( IL_j \) of \( n_j \) are as follows:

\[
IL_i = R_i' / (p_i \alpha_i)
\]

\[
(9)
\]

\[
IL_j = R_j' / (p_j \alpha_j)
\]

(10)

After balancing, the balanced residual lifetime \( BL_i \) of node \( n_i \) and the same \( BL_j \) of \( n_j \) are given by,

\[
IL_i = R_i' / (p_i' \alpha_i)
\]

(11)

\[
IL_j = R_j' / (p_j' \alpha_j)
\]

(12)

Without any loss of generality, please assume that \( p_i > p_j' \). Then, automatically \( p_i > p_j' \) because \((p_i' + p_j') = (p_i + p_j)\).

It means that \( n_i \) gains in terms of residual lifetime whereas \( n_j \) loses it.

\[
\text{Let } (p_i - p_j') = (p_i - p_j) = c
\]

(13)

Lifetime gain \( G_i \) of \( n_i \) is formulated as,

\[
G_i = R_i' / (p_i' \alpha_i) - R_i' / (p_i \alpha_i)
\]

(14)

Therefore, \( G_i = (R_i' / \alpha_i) (1/p_i' - 1/p_i) \)

i.e. \( G_i = (R_i' / \alpha_i) [1/p_i' - 1/p_i + c] \)

(15)

Lifetime loss \( L_j \) of \( n_j \) is formulated as,

\[
L_j = R_j' / (p_j' \alpha_j) - R_j' / (p_j \alpha_j)
\]

(16)

Therefore, \( L_j = (R_j' / \alpha_j) (1/p_j - 1/p_j') \)

i.e. \( L_j = (R_j' / \alpha_j) [1/(p_j' - c) - 1/p_j'] \)

(17)

Since after residual lifetime balancing \( R_i' / (p_i' \alpha_i) = R_j' / (p_j' \alpha_j) \), so for \( G_i > L_j \) to be true, the required condition is,

\[
c/(p_j' - c) > c/(p_j' - c)
\]

i.e. \( 1/p_j > 1/p_j' \) (from (13))

i.e. \( p_i > p_j \)

If \( p_i = p_j \) then the average node lifetime remains unaffected and if \( p_i < p_j \) then the average node lifetime decreases. During balancing, the average node lifetime should not suffer.

### 2.5 Pair-wise Balance of Energy Depletion

After balancing, let \( p_i' \) and \( p_j' \) denote forwarding load of \( n_i \) and \( n_j \).

So, \((p_i' + p_j') = (p_i + p_j) \)

Energy depletion rates of \( n_i \) and \( n_j \) are given by \((p_i' \alpha_i)\) and \((p_j' \alpha_j)\) respectively. For balanced energy depletion, these two must be made equal. So,

\[
p_i' \alpha_i = p_j' \alpha_j
\]

(18)

\[
\text{So, } p_i' = p_j' \alpha_j / \alpha_i
\]

(19)

Putting this in (3) we get,
\[ p'_j = (p_i + p_j) \alpha_j / (\alpha_i + \alpha_j) \]  
(20)

If \( p'_j \) is a fraction, we take, \( p'_j = \lceil p'_j \rceil \). Putting this in (3) we get,
\[ p'_j = (p_i + p_j - p'_j) \]

The utility of energy depletion rate balancing is that it arrests further deterioration of difference of residual energy of \( n_i \) and \( n_j \).

The proof is given in the appendix section.

2.6 Effect of Balancing Energy Depletion on Average Node Lifetime

In this case also, the expressions for \( G_i \) and \( L_j \) remain same as in section 2.3. Since after energy depletion balancing, \( (p'_j, \alpha_j) = (p'_j, \alpha_j) \), so for \( G_i > L_j \) to be true, the required condition is,
\[ c R'_i / (p'_j + c) > c R'_j / (p'_j - c) \]

i.e. \( (R'_i / p'_j) > (R'_j / p'_j) \)  
(21)

If \( (R'_i / p'_j) = (R'_j / p'_j) \) then the average node lifetime remains unaffected and if \( (R'_i / p'_j) < (R'_j / p'_j) \) then the average node lifetime decreases. During balancing, the average node lifetime should not suffer.

III. HOW ALTERNATIVE NODES HELP IN ROUTING

Let \( n_j \) be an alternative of \( n_i \). Without any loss of generality we can assume that \( n_j \) channels some of its load through \( n_i \) i.e. \( p'_i < p_i \) and \( p'_j > p_j \). Balancing will be possible if the message queue of \( n_j \) can handle the extra load i.e. \( (M_j - W_j - (p'_j - p_j)) \geq 0 \). Accordingly \( n_j \) instructs some of its uplink neighbours to canalize their packets through \( n_j \) now instead of \( n_i \). If the link from an uplink neighbour to \( n_i \) breaks, that neighbour now forwards the packet destined to \( n_i \) to \( n_j \) now, instead of initiating a new route discovery session, saving a huge amount of message cost.

Preventing route discovery during link repair saves a huge amount of message cost.

**Proof:** With the initiation of a new route discovery session, route request packets are broadcast in the network which traverse at least 1 and at most \( H \) hops (i.e. \( (1+H)/2 \) hops on an average) where \( H \) is the maximum allowable hop count in the network. Please assume that, on an average, the number of downlink neighbours of a node is \( q \). So, on an average, the number RR of route request packets generated is given by
\[ RR = q + q^2 + q^3 + \ldots + q^{(H+1)/2} \]

i.e. \( RR = q(q^{(H+1)/2} - 1)/(q - 1) \)  
(22)

Preventing the injection of RR amount of route request packets into the network in the context of repairing each broken link, is a huge one. This reduces message collision in ANB embedded protocols increasing the data packet delivery ratio.

Repairing the link through a balanced alternative protects the energy efficiency of the path. Actually, alternative nodes are a ready and effective solution to the link breakage problems. Improvement in average node lifetime in ANB produces more alive nodes at any point of time in the network as far as ANB-embedded protocols are concerned compared to their ordinary versions. The utility of ANB embedded routing protocols is very high in today’s dense networks where alternative nodes are easily available.

IV. SIMULATION RESULTS

Simulation of the mobile network has been carried out using ns-2 [12] simulator on 800 MHz Pentium IV processor, 40 GB hard disk capacity and Red Hat Linux version 6.2 operating system. Graphs appear in figures 2 to 7 showing emphatic improvements in favor of limited area route discovery. Number of nodes has been taken as 20, 50, 100, 300 and 500 in different independent simulation studies. Speed of a node is chosen as 5m/s, 10 m/s, 25 m/s, 35 m/s and 50 m/s in different simulation runs. In the simulation runs where speed is varied, number of nodes is kept constant at 300. Similarly, when number of nodes is varied, the speed is kept constant at 25 m/s. Transmission range varied between 20m and 100m. Used network area is 500m x 500m. Used traffic type is constant bit rate. Mobility models used in various runs are random waypoint, random walk and Gaussian. Performance of the protocols AODV, ABR and FAIR are compared with their ANB embedded versions ANB-AODV, ANB-ABR and ANB-FAIR respectively. In order to maintain uniformity of the implementation platform, we have used ns-2 simulator for all the above-mentioned communication protocols. The simulation matrices are data packet delivery ratio (total no. of data packets delivered×100/total no. of data packets transmitted), message overhead (total number of message packets transmitted including data and control packets) and alive node ratio (total no. of alive nodes×100/total no. of nodes in the network). Simulation time was 1000 sec. for each run.
Figure 2: Data packet delivery ratio vs number of nodes

Figure 3: Data packet delivery ratio vs node speed

Figure 4: Cost of messages vs number of nodes

Figure 5: Cost of messages vs node speed

Figure 6: Alive node ratio vs number of nodes

Figure 7: Alive node ratio vs node speed

Figure 2 shows that the initially the data packet delivery ratio improves for all the protocols with increase in number of nodes and then it starts reducing. The reason is that the network connectivity improves with increase in number of nodes, until the network gets saturated or overloaded with nodes. When the overloading occurs, cost of messages become very huge and the packets hinder one
over AODV is more than that produced by ANB-ABR over AODV. So, performance enhancement of ANB-AODV is less devastating in ABR and FAIR than in AODV. Please note that, the phenomenon like route discovery and link intensity determined by the logic of the protocol itself. Actually, link breakage in all protocols increases message generation of a huge number of RREQ packets once again. discovery session is initiated altogether which requires injection into the network to repair the links broken due to node exhaustion. This reduces the packet collision. As a result, data packet delivery ratio of ANB embedded versions of the above-mentioned protocols also increase compared to the ordinary versions of those. The improvements are evident from figures 2 to 7.

Figure 4 shows that for all the protocols cost of messages increase with increase in number of nodes. This is quite self-explanatory. From figure 6 it may be seen that as the number of nodes increase, the alive node ratio decreases. The reason is that more number of communications is initiated with increased number of nodes and due to better network connectivity more destinations can be tracked now which were initially disconnected. This, along with the phenomenon of more packet collision increases the energy consumption in nodes reducing the alive node ratio. Figures 3, 5 and 7 are concerned with the influence of node speed on these metrics. As the node speed increases, many new links form and older ones break increasing the network congestion and message collision. Colliding messages are unable to reach their respective destinations; hence they need to be retransmitted. This causes injection of some more route-request messages. As a result, packet delivery ratio and alive node ratio decreases with increased cost.

ANB improves the average lifetime of network nodes. Automatically it reduces the occurrence of link breakage due to node exhaustion and tremendously contributes to avoid network partition. This not only improves alive node ratio of ANB-embedded routing protocols compared to their ordinary versions, but also significantly reduces the number of route-request messages that would have been otherwise injected into the network to repair the links broken due to node exhaustion. This reduces the packet collision. As a result, data packet delivery ratio of ANB embedded versions of the above-mentioned protocols also increase compared to the ordinary versions of those. The improvements are evident from figures 2 to 7.

V. CONCLUSION

The concept of alternative node based balancing (ANB) presented in this paper greatly reduce message overhead of the network by increasing the average node lifetime. As a result, data packet delivery ratio increases along with alive node ratio with the decreased message cost. Also the phenomenon like network partitioning can also be avoided up to a great extent in ANB-embedded protocols. In today’s dense network, the utility of ANB embedded protocols are more applicable because it increases the chance of finding alternative nodes.

REFERENCES

Appendix

Here the following two propositions are proved.

**Proposition 1:** The alternative relation is non-reflexive, symmetric and transitive.

**Proof:** As per the definition of alternative node, since a node cannot be its own alternative, so the alternative relation is non-reflexive.

In order to prove the symmetric property, let \( n_j (n_j \neq n_i) \) be an alternative node of \( n_i \) at time \( t \). So,

i) \( U_i(t) - \{ n_j \} = U_j(t) - \{ n_i \} \)

ii) \( D_i(t) - \{ n_j \} = D_j(t) - \{ n_i \} \)

Interchanging the L.H.S. of the above equations by their respective R.H.S. we get,

i) \( U_j(t) - \{ n_i \} = U_i(t) - \{ n_j \} \)

ii) \( D_j(t) - \{ n_i \} = D_i(t) - \{ n_j \} \)

These are the conditions for \( n_i \) to be an alternative of \( n_j \) at the same time \( t \). So, the relation is symmetric.

In order to prove the transitivity property, let \( n_j \) be an alternative of \( n_i \) and \( n_k \) be an alternative of \( n_j \) at time \( t \). Then,

i) \( U_i(t) - \{ n_j \} = U_j(t) - \{ n_i \} \) \hspace{1cm} (23)

ii) \( D_i(t) - \{ n_j \} = D_j(t) - \{ n_i \} \) \hspace{1cm} (24)

and

i) \( U_j(t) - \{ n_k \} = U_k(t) - \{ n_j \} \) \hspace{1cm} (25)

ii) \( D_j(t) - \{ n_k \} = D_k(t) - \{ n_j \} \) \hspace{1cm} (26)

Subtracting \( \{ n_k \} \) from both sides of (13), we get

\[
U_i(t) - \{ n_j \} - \{ n_k \} = U_j(t) - \{ n_i \} - \{ n_k \}
\] \hspace{1cm} (27)

Replacing \( (U_i(t) - \{ n_k \}) \) by \( (U_i(t) - \{ n_j \}) \) in (17), \( U_i(t) - \{ n_j \} - \{ n_k \} = U_j(t) - \{ n_i \} - \{ n_k \} \) \hspace{1cm} (28)

Cancelling \( \{ n_k \} \) from both sides of (28) we get,

\[
U_i(t) - \{ n_j \} = U_j(t) - \{ n_i \}
\] \hspace{1cm} (29)

Similarly subtracting \( \{ n_k \} \) from both sides of (24), we can prove that

\[
D_i(t) - \{ n_j \} = D_j(t) - \{ n_i \}
\] \hspace{1cm} (30)

If (29) and (30) are true, then we can say that \( n_k \) is an alternative of \( n_i \).

**Proposition 2:** The utility of energy depletion rate balancing is that it arrests further deterioration of difference of residual energy of \( n_i \) and \( n_j \) where \( n_j \) is an alternative of \( n_i \).

**Proof:** Let at time \( t \), the residual energies of \( n_i \) and \( n_j \) are \( R_i \) and \( R_j \) respectively. Also assume that after time interval \( \tau \), their residual energies will be \( R_{i1} \) and \( R_{j1} \) respectively. Then,

\[
R_{i1} = R_i - p_i \alpha_i \tau
\]

\[
R_{j1} = R_j - p_j \alpha_j \tau
\]

So, \( R_{i1} - R_{j1} = R_i - R_j - p_i \alpha_i \tau - p_j \alpha_j \tau \) \hspace{1cm} (33)

Since energy depletion rate is balanced, so \( p_i \alpha_i = p_j \alpha_j \)

Putting this in (33) we get

\[
R_{i1} - R_{j1} = R_i - R_j
\] \hspace{1cm} (34)

The equation (24) indicates that after energy depletion rate balancing, the difference of residual energy of \( n_i \) and \( n_j \) does not deteriorate.

Authors Biography

Dr. Anuradha Banerjee is an assistant professor in the department of Computer Applications at Kalyani Govt. Engg. College. She completed her B.E. from Bengal Engineering College Sibpur in 2001 and Ph.D. from West Bengal University of Technology in 2013. Her research areas are Ad hoc network, Soft computing and Neural networks. She has 22 publications in different international and journal journals.

Dr. Paramartha Dutta did his Bachelors and Masters in Statistics from the Indian Statistical Institute, Calcutta in the years 1988 and 1990 respectively. He afterwards completed his Master of Technology in Computer science from the same Institute in the year 1993 and Doctor of Philosophy in Engineering from the Bengal Engineering and Science University, Shibpur in 2005 respectively. He has served in the capacity of research personnel in various projects funded by Govt. of India, which include Defence Research and Development Organization, Council of Scientific and Industrial Research, Indian Statistical Institute, Calcutta etc. Dr. Dutta is now a Professor in the Department of Computer and system Sciences of the Visva Bharati University, West Bengal, India. Dr. Dutta is a Life Fellow of the Optical Society of India (OSI), Life Member of Institution of Electronics and Telecommunication Engineers (OSI), Computer Society of India (CSI), Indian Science Congress Association (ISCA), Indian Society for Technical Education (ISTE), Indian Unit of Pattern Recognition and Artificial Intelligence (IUPRAI) - the Indian affiliate of the International Association for Pattern Recognition (IAPR), Member of Associated Computing Machinery (ACM), IEEE, Computer Society, USA.