Biased random algorithm for load balancing in Wireless Sensor Networks (BRALB)

Barra Touray & Princy Johnson

Abstract— A Wireless Sensor Network (WSN) consists of large number of small, inexpensive nodes that depend on their sensors, transmission and routing capabilities to collect and disseminate critical data. The WSNs can be used for various application areas (e.g., Structural monitoring, agriculture, environment monitoring, Machine health monitoring, military, and health). For each application area there are different technical issues and remedies. Routing algorithm is a very important component in WSNs. Most of the time, the nodes in a WSN run on battery with limited power. Hence the need for an energy efficient routing algorithm is becoming a critical issue for WSNs. In this paper a Biased Random Algorithm for Load Balancing (BRALB) in Wireless Sensor Networks for environment monitoring is proposed. It is based on energy biased random walk. It does not require any global information. It uses probability theory to acquire all the information it needs to route packets based on energy resources in each node. It is shown in this paper by using both statistical method and simulation that BRALB uses the same energy as the shortest path first (spf) routing in cases where the message to be sent is comparatively small in size, with the inquiry message among the neighbors. It is also shown to balance the load (i.e. the packets to be sent) among the neighboring nodes. So, the major advantage of the proposed algorithm over the shortest path first routing is that BRALB does not result in network partitioning as the spf due to repeated use of routes in the network.

Index Terms— biased random walk, routing algorithm, shortest path first, wireless sensor network, energy-efficient, load balancing, statistical method.

1 INTRODUCTION

A Wireless Sensor Network (WSN) is made up of several sensor nodes with combined wireless communication and minimal computation facilities onboard along with sensing of physical phenomenon which can be embedded in a physical environment [1], [2]. A typical sensor node has the following basic components: a sensing module, a processing module, a transceiver, and a power supply. These tiny sensor nodes are densely deployed and are used in various applications such as environmental monitoring, target tracking, military, habitat sensing, danger alarm and medical analysis [1]. One of the key issues in WSNs is the data delivery scheme between sensors and the data collection unit called the sink. The sink serves as the gateway between the network and the end users. The sink is a special node that is reliably connected by wire or satellite to the Internet and has adequate power supplies and processing power.

- *Barra Touray is currently pursuing PhD degree program in engineering at Liverpool John Moores University, Liverpool , L3 3AF. E-mail: b.touray@2010.ljmu.ac.uk barra2ray@yahoo.co.uk .*
- *PrincyJohson is senior lecturer at Liverpool John Moores University. E-mail: p.johnson@ljmu.ac.uk .*

The other sensor nodes are so tiny that their energy supply, storage space, data processing and communication bandwidth are very limited, and therefore every possible means of efficient usage of these resources is aggressively sought. In WSNs the energy of the node is an invaluable commodity and hence various schemes such as the WSN topology, routing algorithms or MAC protocols have been used to maximize the network lifetime [3],[4],[5]. The sources of energy consumption are computation and communication. The communication between nodes is the major source of energy consumption in WSNs and hence it is the most expensive aspect. In this paper, novel techniques for routing algorithm are proposed for the maximization of the network life time.

The size of the message routed in WSNs is application dependent. For this reason sensed data are classified according to size namely: large-size, medium-size and small size data. BRALB is intended for applications that use small size data such as in environmental monitoring. In such applications the message to be sent is in comparative size to the inquiry message between the neighboring nodes. A WSN consists of a large number of nodes that are densely deployed in random fashion and exploring their best possible use is a challenging problem. For the sensors to adequately monitor some physical quantity in a given area, the sensors need to cover the area without leaving any void or un-sensed area. Hence, the need for large node deployment since the sensing range is limited and the fact that the nodes are randomly deployed. Because of the large number of nodes deployed and the limited energy and computational power available to the nodes in WSNs, it is prohibitive to use routing protocols that require global information regarding the whole network.

In BRALB a routing protocol based on random walk for a square grid based network topology has been proposed. This routing algorithm does not require any global location information and achieves an inherent load balancing property for WSN, which is difficult to achieve with any other routing protocol. In this paper the probability of successful transmissions from the source to the destination by random walk was analyzed statistically, and it is proved that random walk is as energy efficient as the shortest path routing algorithm while avoiding network partitioning due to repeated use of same routes in the network. The qualifying assumption is that the message to be sent is small in size compared to the inquiry message among the neighbor nodes. BRALB uses probability to enable the nodes to make a fair guess about the energy level of their neighboring nodes and select the node with the maximum energy to forward the message [6].

The rest of the paper is organized as follows. In section 2 a review of the related work is presented while in section 3 the proposed BRALB algorithm is discussed. The statistical and simulation analyzes of BRALB are presented in sections 4 and 5 respectively. Section 6 concludes the paper.

2 RELATED WORKS

In recent years, WSNs have been attracting rapidly increasing research efforts [8], [9], [2]. Research into routing protocols has been the major highlight of WSN research. The traditional routing schemes proved unsuitable to adapt to WSNs due to the specific characteristics of WSNs and hence many new algorithms that can support WSNs [10], [11], [12], [13], [14] have been developed.

The ant algorithms family stem from the behaviour of ants which communicate with each other by the use of a chemical substance called pheromone. The ants release pheromone on their path and these influence other ants to follow the same path. At the beginning there is no pheromone on the branches and ants have no clue about the length of branches. Also, in the beginning there might be several paths leading to the destination, but as time goes by the shortest path will have more pheromone. The more the pheromone laid by the ants on a path larger the probability that they will visit the path next time. Thus there is a positive feedback in the groups of ants.

In [16] Roth and Wicker present an algorithm that closely relates to simple ant routing algorithm (ARA) by using the same distance vector principles with some difference in terms of route discovery and failure recovery. This algorithm requires the use of hello messages to initialise the links between neighbors. Once this is done the hellos are never sent again unless when a nodes routing table is empty. In termite a node discovers a route by sending a RREQ packet. This message walks in a random fashion and the subsequent hops created in each stage before reaching its destination or being deleted is done in a uniformly distributed random decision function. When a RREQ packet finds a node that knows how to get to the target destination, the node generates a RREP packet and sends it back to the source. In this way packets are sent from source to destination. The RREP packet follows the pheromone laid by the RREQ, and further strengthening the pheromone along that path, thereby creating a clear path for the data packet to take.

Minimum Cost Forwarding Algorithm (MCFA) [11] is based on the fact that the destination of a packet is already known and it is the Base Station (BS). Hence the routing is always done toward the base station. The sensor nodes (SN) do not need to have a unique ID or to maintain a routing table. Since the routing direction is already known this reduces the number of packets transmitted within the network and so maximises the network life time. A node just needs to maintain a least cost path from itself to the BS. Messages that are forwarded by a SN are also broadcast to its neighbors by the SN. A receiving node always checks whether it is along the least cost path between the source SN and BS. If it is then it rebroadcasts the message to its neighbors. This is repeated until the message gets to the BS. The process of each sensor node acquiring the knowledge of the least cost path between itself and the base station is as follows. At the initial stage each node sets its least cost path to the base station to infinity. The BS then broadcasts a message to all the nodes in the network with the cost set to zero. A receiving node verifies whether the estimate in the message plus the link cost on which the message was received is less than the current estimate. If it is true, the current estimate and the estimate in the broadcast message are updated accordingly and the message is then re-broadcast, otherwise nothing happens. There is a potential problem here in that nodes farther away from the BS will get more updates and some nodes may have multiple updates. The MCFA is modified with the back-off algorithm to solve this problem in [11].

In [13] an energy efficient routing protocol called Remaining-Energy Based Routing (REB-R) is proposed and compared to two well known protocols AODV and T-ANT using NS-2. The main idea of REB-R is the broadcasting of nodes' remaining energy along with the data in the data packet. In order to do this, two short packets types are created namely FWD_ROUTE and DATA. After the deployment of the sensor nodes the sink then broadcasts FWD_ROUTE packet to all its neighbors. A node that received this packet stores it as FIRST_TTL in its TTL (time-to-live). This node will then broadcast a fresh new FWD_ROUTE packet with TTL incremented by one. Node use TTL value to form neighbors, this value also shows how far a node is from the sink. Level-one nodes are neighbors to sink while level-two nodes are neighbors to level-one nodes and so on. When a node received a FWD_ROUTE packet with a smaller TTL or equal to its FIRST_TTL it is recorded as nominee for the parent node of the receiving node. If the TTL is more than FIRST_TTL then the packet is dropped to avoid broadcasting unnecessary information. During this period the sender of any packet that has minimum TTL and maximum energy is selected as the current parent node. When the network is steady, current parent node is selected from the list of nominees which has the maximum energy. After this the nodes will have enough information to route packets to the sink. Though this algorithm performs better than AODV and T-ANT, it has lot of overheads and might deplete the nodes' energy when compared to BRALB.

In [13] an advanced ant colony algorithm based on cloud model (AACOCM) which is based on ant colony optimization algorithm (ACOA) is proposed. AACOCM is multi-objective bandwidth constraint algorithm and it aims to find a route in the network which has sufficient resources to optimize some network parameters such as data packet lost rate, delay and energy consumption. This algorithm introduces a multi-objective evaluation function by putting into account the three mentioned parameters. Three ant types are defined namely: ordinary ants, greedy ants and usual ants with different behavior which help to achieve the multi-objective inspection and evaluation. The ants walk from source to destination in a gradual routing tree formation.

In [14] a routing protocol for WSN based on the PEGASIS protocol using an improved ant colony algorithm instead of the greedy algorithm to construct the chain is proposed. This protocol is called PEG-ant. In the chain construction process in order to choose a node as the next one on the chain, all the current node's neighbors are candidates whereby the selection criteria are the remaining energy of the candidate, the amount of energy consumed when a unit data is transmitted along the branch between the current node and the candidate, and the pheromone quantity in a branch. It forms a chain that makes the path more evenly-distributed when compared to PEGASIS. A leader is selected for each round of transmission to directly communicate with the BS based on node with maximum current energy. Along the built chain and in the direction of leader, starting from further nodes, each node fuses the received data with its own as one packet to transmit to the other neighbor. This will eventually reach the leader. The leader then sends the final fused data to the BS.

In [15] PERA algorithm uses Ant-like agents to discover and maintain paths in MANETs. At the initial stage there are no routing tables or no next hop from source to destination. The initialization and neighbour discovery is done by single-hop, broadcast HELLO messages that are transmitted periodically in order to build the neighbour list. During the initial stages the first forward ant is sent by a node to its neighbors with equal probability to any of its neighbors. If a node has N neighbors the probability of selecting each neighbour as the next hop is $\mathbf{1/N}$. These probabilities will change from being uniform as they are adjusted when the source received the backward ants from the destination. Therefore a node will have different probabilities for forwarding to the next hop neighbour as time goes on due to the backward ants' feedback.

Each neighbour has a routing table in this format (Destination, Next hop, Probability). In addition each node also maintains a table of statistics containing mean and variance for each destination 'd' to which a forward ant has been previously sent. These routing tables are built and maintained by the information gathered from the Forward ants, Uniform ants, Regular ants and the Backward ants.

The forward ants are agents generated periodically by a node and sent to randomly chosen destination so as to gather routing information. A node that sends a forward ant creates a routing table if there is none for that node; the intermediate nodes also do the same thing. A forward ant packet contains: source, next hop and destination IP addresses, Stack and hop count. The Stack of the forward ants is a dynamically growing data structure that lists the IP addresses of the traversed nodes as well as the time. Forward ants faced the same network condition as they are routed just like normal data packets. The forwarding of the forward ant is probabilistic and provides exploration of paths available in the network. These ants were then called Regular Ants so as to distinguish them from Forward Uniform Ants. To prevent loop any forward ant that went to a node and see that the nodes IP is in its Stack then the forward ant is destroyed.

In order to promote the discovery of new routes the authors in [15] created a uniform ant. These ants are created in the same manner like regular ants but are routed differently. The routing of the uniform does not use the routing table it uses equal probability to select the next hop node. These ants help to explore and reinforce newly discovered routes and prevent the saturation of previously discovered paths.

A backward ant is created when a forward ant reaches its destination. The backward ant inherits the Stack of the forward ant; it uses this to quickly update the source node and all the intermediate nodes along the path by using the high priority queue.

PERA was compared to AODV, and the results indicate that it performs better than AODV in terms of delay.

The use of random walk in WSNs has been extensively researched [7], of which the Directional Rumor Routing in Wireless Sensor Networks is based on the random walk of agents. The aim of this algorithm is to improve the latency and energy consumption of the traditional algorithms using propagation of query and event agents in straight lines, instead of using purely random walk paths. Directed Rumour Routing has two phases for calibration. In the first phase each node sends a Hello message stating its position to each of its neighbor. The hello messages are used by the receiving nodes to record their neighbors and their positions. During the second phase each node tests whether it is at the edge of the network. When a node senses an event, it creates a number of event agents and propagates them into the network along some linear paths forming star-like propagation trajectories. These event agents are not allowed to pass the edge nodes. After this a node is randomly chosen as the sink node. The sink node creates some query agents for each fired event. Each agent contains the id of the current node, the id of the previous node (depicting the direction of events), location information of the source node, and a table containing the ids of the events and distances to them. The disadvantage with this method is that the hello messages drain the WSN of its limited bandwidth and imposes additional energy drain on the nodes. However, in BRALB the need for hello messages has been eliminated.

Sensing and connectivity are vital parts of WSN. For efficiency it is important to use a minimum number of sensors possible while maintaining the connectivity to cover a sensing area in order to reduce cost. The connectivity requirement is to ensure that all the nodes are able to communicate with the sink either through single or multi-hop communication, while the sensing requirement is to make sure that the entire WSN is at least within the sensing area [6].

Various topologies have been proposed to address both coverage and connectivity requirements. If the region to be covered is in 2-dimensional plane there are three types of regular topology for WSNs namely: square, hexagon, and triangle-based topologies. It has been demonstrated in [6] that among those three topologies, triangle-based topology provides the best sensing strength and reliability while trading off energy consumption and total coverage. While WSNs in hexagon-based topologies provide maximal connected-coverage given the same number of nodes, Square-based topology's performance lies between the two and yet is the simplest architecture. It is for this reason that the analysis of BRALB is based on a square grid based network topology.

3 Biased Random Algorithm for Load Balancing (BRALB)

BRALB is implemented on a square grid based network topology with each node having four neighbors except for the border nodes.

In BRALB a node will have a message counter for each neighbor. When node A sends a message to say node B, then node A will increase the counter designated for node B by a value of 1. On the other hand the receiving node B will update its counter designated for node A; increasing it by a value of 4. The reason for the four step increment for a message received is to try and predict the message sent by that particular neighbor. It is a known fact in statistical studies that when one node has an equal chance of selecting four neighbors to send some messages then after some time if one of those neighbors received say 5 messages then it is reasonable to assume that the remaining three neighbors may as well get 5 messages each. In this way it is a fair guess to say that the sending node must have sent 20 messages in all. By this way for every message received from a neighbor one can safely assume that the neighbor may have sent 4 messages hence the four step increment. In this way all the messages sent and received by a neighbor can be predicted and therefore the energy level of that particular node can be predicted as there is a correlation between node energy and messages processed by that node. Therefore, when node A with four neighbors wants to send a message to the sink which is say six hops away from itself, it will first inspect its counters and select the node with the least count. This process is repeated until the message reaches the Sink. In this paper it has been proved statistically that in application whereby the message to be sent to the Sink is comparable to the size of the inquiry message then this routing mechanism can route to the Sink by using the same energy as the SPF algorithm.

The messages sent and received represent the energy used in the network and therefore by biasing the nodes to forward to the nodes with fewer messages (both sent and received) the network load balances the energy of the network. This will avoid using the shortest path all the time by distributing the energy usage fairly within the network and hence will avoid partitioning of the network.

4 Statistical analysis of BRALB

BRALB can be statistically analyzed on a squared grid based network topology by investigating the probabilities of forwarding a packet through the squared grid based network topology using the algorithm.

When a node wants to report an event to the sink or base station, in WSNs, it would usually contact all it neighbors and then forward it to the neighbor with the least number of hops to the sink. In applications where the message to be sent is small in size such as abnormal activity detection system and danger alarm system, the communication cost between neighbor pairs for choosing the next-hop neighbor is comparable with that of transmitting the real message [6], [17] itself. In such traditional WSNs topologies, a node needs to communicate with three of its neighbors before forwarding the message. It will be further assumed that the communication cost between two nodes for next-hop inquiry is equal to the transmission cost of the real message to be sent, and denote it by one energy unit 'e'. When a source node wants to send a packet to the sink it will contact its four neighbors before selecting the next hops. The energy used by the source to make this possible is therefore four units of energy plus the extra unit for sending the actual message.

For the next-hop nodes they will send a response to the sender to verify that they have a valid route to the destination after contacting their three neighbors. The message is then transmitted to this node which then forwards it to its chosen neighbor. Therefore the total energy for routing data from one next-hop node to its next-hop node is the sum of energy required for responding to route request (1e), the energy required for contacting its three neighbors (3e) and the energy required for transmitting the real data (1e), which totals to 5e.

However, the cost of routing data to the next hop in random walk is just one unit of energy 'e'. This is because in BRALB the data is just forwarded at random to any of the three neighbors without any initial inquiry. If the data is to be sent from a source say G to the sink Y which is ten hops away then the energy cost would be 50e for the shortest path routing and only 10e for BRALB if it routes the message itself in 10 hops.

In BRALB when a node receives a packet the next hop is determined by the number of packets it had already received or sent to its neighbors. Based on this it will select one neighbor and forward the packet to the neighbor who has either sent or received the least number of messages. Therefore, the energy cost for one hop transmission using the proposed algorithm is only one unit of energy 'e', while the energy cost for one hop of data transmission using the traditional shortest path is five units of energy (5e), including the inquiry stage. The shortest path routing protocol uses the least amount of energy in routing a message from source to the destination. The effectiveness of BRALB in terms of energy usage is proved to be better under certain conditions than that of the shortest path routing protocol. The probability of successfully sending data from the source to the sink will be analyzed in order to determine the effectiveness of BRALB when used in a squared grid based network topology.

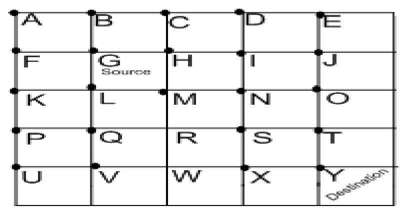

Fig. 1. A square-based WSN topology

The Fig. 1 above illustrates an example of sensor network consisting of 25 nodes named from A to Y arranged in a squared grid based network topology. In this network, node G is designated as the source and node Y acts as the sink/destination. The routing scenario where a node cannot forward a message to the node it received the packet from will be assumed. So, only the source node will have four neighbors to select from, and the rest of the nodes along the path will have only three neighbors to choose from as long as they are not border nodes. Therefore it is fair to assume that on an average all nodes will have three neighbors to select from.

In a square grid based network topology, the expression for the relationship between the boundary nodes and the total number of nodes can be developed as given below. Let the total number of nodes in the grid be represented as $\boldsymbol{m * m}$ and the number of boundary nodes be represented by b. Then the total number of boundary nodes in any given square grid can be written as:

$$\boldsymbol{b = m + m + (m-2) + (m-2)}$$

$$\boldsymbol{b = 4m - 4} \tag{1}$$

Then the ratio between the boundary nodes and the total number of nodes is

$$\frac{\boldsymbol{4m-4}}{\boldsymbol{m*m}} \tag{2}$$

The decimal value 4 can be ignored as $\boldsymbol{m} \to \infty$

$$\mathbf{lim}_{m\to\infty} \frac{4m}{m^2} = \frac{4}{m} \to \tag{3}$$

Hence it is reasonable to assume that the non-boundary nodes are negligible as the grid gets larger.

In figure 1 if node G sends a packet to either F or B it is in the wrong direction, whereas if it sends to either H or L it is in the correct direction towards the sink Y. Using this condition we can calculate the probability of a data packet being successfully sent from node G to the sink node Y. The sink node Y is six hops away from the source node G.

In this scenario it will be assumed that all the neighbors have equal chances of sending or receiving from a neighbor. Therefore the chances of selecting any of the three neighbors are the same. The probability that a packet is forwarded in the wrong or correct direction depends on whether a forwarding node has received the packet from the wrong or correct direction.

For the packet to be successfully routed from node G to node Y, using the shortest path over six hops, it must be forwarded in the right direction for all the hops. The probability that this happens is calculated as follows. If a node received a packet from a correct direction then the probability of forwarding in the correct direction is calculated as follows. A node has three neighbors to select as next-hop. Two of these neighbors are in the correct direction and have a probability of $\frac{2}{3}$ for forwarding in the correct direction while the remaining neighbor is in the wrong direction and has a probability of $\frac{1}{3}$ for forwarding in the correct direction. At the initial stage when a neighbour has three messages to send, BRALB will first pick one of its neighbours at random and will forward the first message. For the second message it will pick one of the other two neighbours as the next-hop while the last message is sent to the remaining neighbour. This process is repeated for every three messages a node has to send. Hence the total probability of forwarding a packet in the correct direction when it has been received from the correct direction is the average of the three neighbors' probabilities $\frac{1}{3}.\frac{2}{3}+\frac{1}{3}.\frac{2}{3}+\frac{1}{3}.\frac{1}{3}=\frac{5}{9}$. If the packet is received from the wrong direction then two of its neighbors are in the wrong direction and only one neighbor is in the correct direction. The two neighbors in the wrong direction have a probability of $\frac{1}{3}$ each of forwarding in the correct direction while the remaining neighbor is in the correct direction and has a probability of $\frac{2}{3}$ forwarding to the correct direction. Hence the total probability of forwarding a packet that's received from the wrong direction to the correct direction is $\frac{1}{3}.\frac{1}{3}+\frac{1}{3}.\frac{1}{3}+\frac{1}{3}.\frac{2}{3}=\frac{4}{9}$. If a node received a packet from a correct direction then the probability of forwarding in the correct direction is $\frac{5}{9}$ and the probability of forwarding in the wrong direction is $\frac{4}{9}$. On the other hand if a node received a packet from a wrong direction then the probability of forwarding in the correct direction is $\frac{4}{9}$ and the probability of forwarding in the wrong direction is $\frac{5}{9}$.

If the packet is to be forwarded at six hops to reach the destination along the shortest path then it must be forwarded to the correct path in six hops and to the wrong path in zero hops. The probability of this happening is shown below:

$$\boldsymbol{p\{d=6\}} = \binom{6}{0}\frac{5^6}{9^6} = \mathbf{0.029401} \quad (4)$$

Where p is the probability that a packet is forwarded and d is total number of hops the packet travels.

However, it is very likely that the packet will not be forwarded to the shortest path by random walk. Therefore it is likely that the packet will be forwarded in the wrong direction before getting to the destination. If the packet is sent one hop in the wrong direction it must move one hop backward towards the correct direction. Therefore, for every one hop in the wrong direction two hops must be added to d to get the total number of hops to route the packet to the destination. In this case where the destination is six hops away from the source, if the packet is forwarded one hop in the wrong direction, then the least number of hops to send it back to the destination would be 8. Therefore, the relationship between the total numbers of hops a packet traverses and the number of hops it traverses in the wrong direction is given as:

$$\boldsymbol{d = k + 2.i}, \quad (5)$$

Where d is the total number of hops a packet travels, k the shortest number of hops between the source and destination and i the number of hops the packet is forwarded in the wrong direction. In the above example k = 6, d = 6 + 2.i. and 'i' must be zero and therefore if the total number of hops (d) is six, the packet was forwarded in the correct direction all the time. With the total number of hops being 8, the packet would have been forwarded 7 hops in the correct direction and one hop in the wrong direction. Similarly for d=10, the packet would have been forwarded 8 hops in the correct direction and two hops in the wrong direction. This can be used to easily calculate the probability of successful transmissions at any total number of hops. The probability of successful transmissions at 12 hops is shown below:

$$\boldsymbol{p\{d=12\}} = \binom{12}{3}\left(\frac{9}{12}.\frac{5}{9}+\frac{3}{12}.\frac{4}{9}\right)^9.\left(\frac{9}{12}.\frac{4}{9}+\frac{3}{12}.\frac{5}{9}\right)^9 \quad (6)$$

Therefore the probability of a packet reaching the destination at any given number of hops d in scenario 1 can be written as:

$$\boldsymbol{p\{d=k+2i\}} = \binom{k+2i}{i}\left(\frac{k+i}{k+2i}.\frac{5}{9}+\frac{i}{k+2i}.\frac{4}{9}\right)^{k+i} .\left(\frac{k+i}{k+2i}.\frac{4}{9}+\frac{i}{k+2i}.\frac{5}{9}\right)^{i} \quad (7)$$

$$\boldsymbol{p\{d\le H\}} = \sum_{i=0}^{\frac{H-k}{2}}\binom{k+2i}{i}\left(\frac{k+i}{k+2i}.\frac{5}{9}+\frac{i}{k+2i}.\frac{4}{9}\right)^{k+i}$$

$$\cdot\left(\frac{k+i}{k+2i}\cdot\frac{4}{9}+\frac{i}{k+2i}\cdot\frac{5}{9}\right)^i \quad (8)$$

Hence the probabilities for successful transmission with 6, 24, 40 and 50 hops are calculated by substituting these values for H and substituting k by 6 as in our example as follows:

$$\boldsymbol{p\{d \le 6\} = 0.02940119411} \quad (9)$$

$$\boldsymbol{p\{d \le 24\} = 0.7401902058} \quad (10)$$

$$\boldsymbol{p\{d \le 30\} = 1.014642274} \quad (11)$$

$$\boldsymbol{p\{d \le 40\} = 1.464255858} \quad (12)$$

In equation 12 it can be seen that the successful transmission probability was more than 100%, this is the result of assuming all the nodes were non-boundary nodes. Hence it can be deduced from the above calculation that, using the proposed biased random algorithm BRALB, it has been proved that it guarantees 100% of the packets to be successfully transmitted using the same energy (30e) as the spf algorithm.

The spf routing requires 5k energy cost as calculated above. In [6] it is found that flooding algorithm requires 2m (m – 1) energy cost in a WSN with a grid size of m * m, m being the number of nodes in each edge. Table 1 compares the energy cost of these algorithms by routing messages on the longest path in the grid (k = 2(m-1)).

TABLE 1 Performance comparison of spf, flooding and BRALB

Routing Algorithm	Number of hops	Energy cost
Shortest path	k	5k
BRALB	5k = (5*6) with 100% successful transmission rate	5k
Flooding	K(k/2-1)	K(k/2-1)

5 Simulation Results

NetLogo simulator is used in order to evaluate and compare the proposed BRALB routing algorithm with SPF. The environment used to test the performance of these two algorithms was modeled using NetLogo's graphic design tool in order to simulate a network. By using Netlogo, network parameters were varied in order to study their effect on the overall performance of each algorithm and also to compare them. The simulator facilitates to deploy the number of resource-constrained nodes and their square-grid connectivity. The simulation was run on an m-dimensional node grid with the number of nodes equal to m × m each having four neighbors except the boundary nodes. In the simulation test bed as depicted in Fig. 2, a total number of 81 (m × m = 9 × 9 nodes) nodes with a node connectivity of 4 has been considered. Performance metrics such as energy consumption, end to end delay, packet loss rate and fault tolerance are used to evaluate the performance of the two algorithms.

The algorithms were implemented in the simulator for comparison. Tests were run for 100 time units, which were considered as ticks. For every performance metric 20 samples were taken and the mean calculated. The performance metrics and their comparisons for each algorithm are described below:

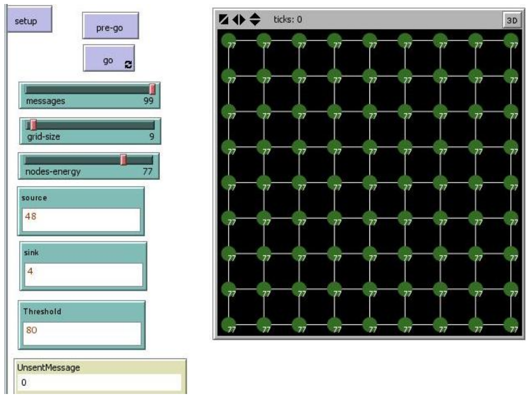

Fig. 2. Simulation snapshot in a grid network for m = 7

A. Energy Consumption

The energy consumption is defined as the total amount of energy used in the network. The total amount of energy is equivalent to the total number of hops the messages traversed. BRALB use almost the same amount of energy as SPF but outperforms SPF in its ability to be more resilient to fault as shown in Fig. 6.

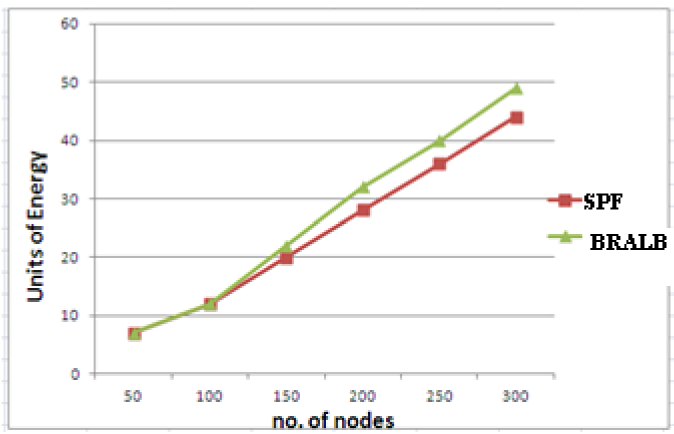

Fig. 3. Total Spent Energy

A. Delay

The delay refers to the time it takes for a message to be routed from the source node to the destination node. The most significant delay in most networks including WSN is the time the devices take before sending the message to the medium of transmission but not the time delay in traversing the medium; as it travels at the speed of light. For this reason the delay is expressed as the number hops in each route as opposed to the actual time it takes to traverse the route.

In order to express the delay in time each hope is taken as a delay of 1ms. In Fig. 4 BRALB has more average delay than SPF which is expected as SPF route to the destination with the least number of hops. For BRALB this is not an issue as it is targeted for applications which are not very sensitive to delay such as environment monitoring.

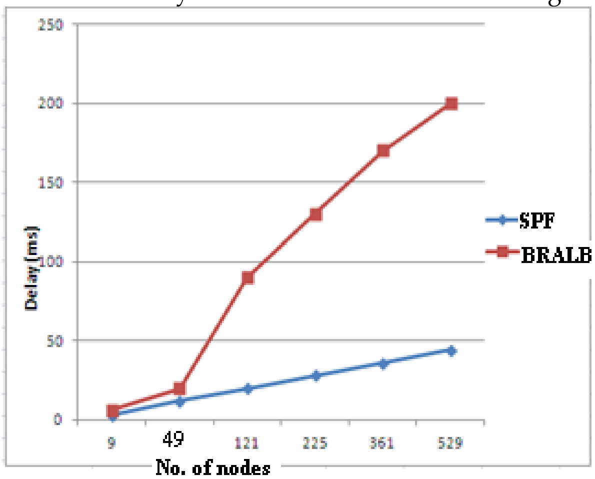

Fig. 4. Average End To End Delay

B. Data packet loss rate

The data loss rate is expressed in terms of the ratio of the total number of messages or packets that reach the destination node versus the total number of messages or packets that were sent from the source node. There are no packet losses for the spf as it has a complete knowledge of the whole network and so can always find the destination as long as there are no dead nodes.

As shown in Fig. 5, BRALB has more packet loss as packets are not allowed to route infinitely. Each packet is calibrated to be discarded once it consumed more energy as an SPF packet would have used. This is called the hop threshold for a packet to live similar to Time-To-Live (TTL) in certain algorithms. However, the packets lost rate here is not significant considering the gain in using limited computational power over SPF were knowledge of the whole network is required.

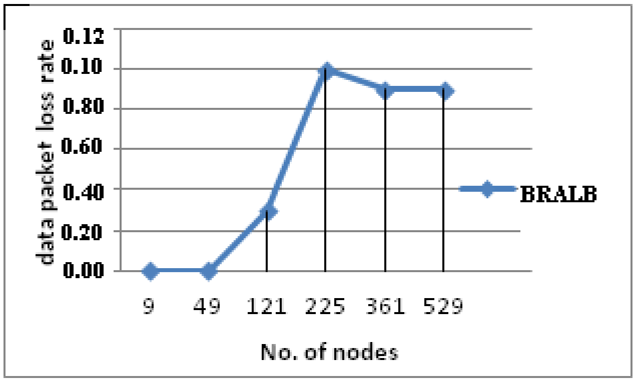

Fig. 5. Data Packet Loss Rate

C. Fault tolerance

Finally in Fig. 6 the robustness of BRALB was tested by randomly failing nodes and it was able to cope with the fault and deliver more than 70% of the traffic as the number of nodes reached 121 and beyond.

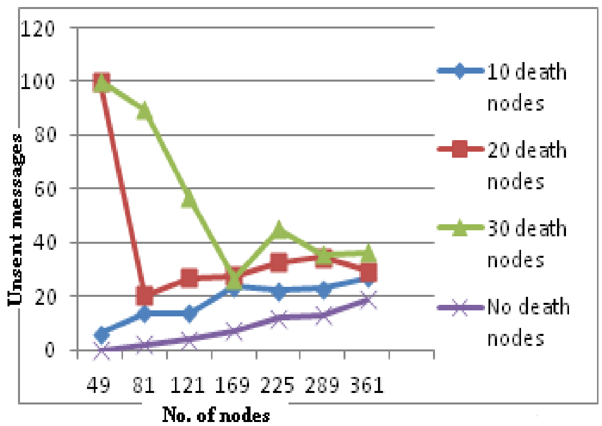

Fig. 6. Fault Tolerance property by BRALB

6 Conclusion

In this paper, a biased random algorithm namely BRALB is proposed for energy-efficient routing in WSNs and the simulation results validate the efficiency of saving energy and prolonging the network lifetime. The work done here has laid the foundation for further complex study of routing in WSNs. The nodes closer to the sink are less efficient in routing to the sink. With this finding a different mechanism is being investigated so as to enable those nodes closer to the sink to be more efficient in routing to the sink. Further research work is being done in order to incorporate clustering to this algorithm to achieve further improvements of the algorithm in order to optimize the utilization of the limited energy in WSNs.

References

[1] I. F. Akyildiz, S. Weilian, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *Communications Magazine, IEEE*, vol. 40, pp. 102-114, 2002.

[2] C. De Morais Cordeiro and D. P. Agrawal, *Ad hoc & sensor networks : theory and applications.* Hackensack, NJ: World Scientific Publishing Co., 2006.

[3] R. Beraldi, R. Baldoni, and R. Prakash, "Lukewarm Potato Forwarding: A Biased Random Walk Routing Protocol for Wireless Sensor Networks," *in Sensor, Mesh and Ad Hoc Communications and Networks, 2009. SECON '09. 6th Annual IEEE Communications Society Conference on*, 2009, pp. 1-9.

[4] D. Vergados and N. Pantazis, "Energy-Efficient Route Selection Strategies for Wireless Sensor Networks," *Mobile Networks and Applications*, vol. 13, pp. 285-296, 2008.

[5] G. Xin, L. Guan, X. G. Wang, and T. Ohtsuki, "A Novel Routing Algorithm Based on Ant Colony System for Wireless Sensor Networks," *in Computer Communications and Networks, 2009. ICCCN 2009. Proceedings of 18th Internatonal Conference on*, 2009, pp. 1-5.

[6] T. Hui, S. Hong, and T. Matsuzawa, "RandomWalk Routing for Wireless Sensor Networks," *in Parallel and Distributed Computing, Applications and Technologies, 2005. PDCAT 2005. Sixth International Conference on*, 2005, pp. 196-200.

[7] H. Shokrzadeh, A. T. Haghighat, F. Tashtarian, and A. Nayebi, "Directional rumor routing in wireless sensor networks," *in Internet, 2007. ICI 2007. 3rd IEEE/IFIP International Conference in Central Asia on*, 2007, pp. 1-5.

[8] M. Jiang, J. Li, and Y.C. Tay, "Cluster Based Routing Protocol functional specification," *IETF Internet Draft*, 1998.

[9] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava, "Coverage problems in wireless ad-hoc sensor networks," *in INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, 2001, pp. 1380-1387 vol.3.

[10] S. Zhen-wei, Z. Yi-hua, T. Xian-zhong, and T. Yi-ping, "An Ant Colony System Based Energy Prediction Routing Algorithms for Wireless Sensor Networks," *in Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference on*, 2008, pp. 1-4.

[11] F. Ye, A. Chen, S. Liu, and L. Zhang, "A scalable solution to minimum cost forwarding in large sensor networks," *In Proceedings of the tenth International Conference on computer Communications and Networks (ICCN)*, pp. 304-309, 2001.

[12] M. Ghane and A. Rajabzadeh, "Remaining-energy based routing protocol for wireless sensor network," *in Computer Architecture and Digital Systems (CADS), 2010 15th CSI International Symposium* , 2010, pp. 67-73.

[13] W. Xu and Z. Li, "The Multi-Objective Routing Optimization of WSNs Based on an Improved Ant Colony Algorithm," *in Wireless Communications Networking and Mobile Computing (WiCOM) 6th International Conference*, 2010, pp. 1-4.

[14] G. Wenjing, Z. Wei, and L. Gang, "PEGASIS Protocol in Wireless Sensor Network Based on an Improved Ant Colony Algorithm," *in Education Technology and Computer Science (ETCS) Second International Workshop*, 2010, pp. 64-67.

[15] J. S. Baras and H. Mehta, "A Probabilistic Emergent Routing Algorithm for Mobile Ad Hoc Networks," *in Proc. of the Conf. on Modeling and Optimization inWireless, Mobile, and Ad Hoc Networks (WiOpt 03*), Sophia-Antipolis, France, March 2003.

[16] M. Roth, S. Wicker, "Termite: Emergent Ad-Hoc Networking", *The Second Mediterranean Workshop on Ad-Hoc Networks*, 2003.

[17] B. Touray, P. Johnson, and M. Randels, "Statistical analysis of Biased Random Energy-Efficient Routing Algorithm (BREERA)," *Workshop on Next Generation Wireless Networks (WoNGeN),2011.*