Roy *et al. Braz J Sci Technol (2016) 3:13*
DOI 10.1186/s40552-016-0026-3

Brazilian Journal of Science and Technology

RESEARCH Open Access

Self-servicing energy efficient routing strategy for smart forest

Sandip Roy*, Rajesh Bose and Debabrata Sarddar

*Correspondence: sandip@klyuniv.ac.in
Department of Computer Science and Engineering, University of Kalyani, Kalyani - 741235 Nadia, West Bengal, India

Abstract

Monitoring the health and population of all forms of wildlife is a challenge for experts. Nature can be unforgiving with incidents of earthquakes, forest fires, tsunamis, and floods appearing without warning. The effects of industrialization take a toll on wildlife as well. As a result, many species of animals have reportedly gone extinct. The practice of installing wireless sensor networks in forests or in remote environments is not new. In this paper, we present a novel approach in deploying static and mobile sensors in forests. The sensors are distributed at random locations throughout a forest region to cover the maximum area possible. Data collected by these sensors is uploaded to cloud for further analysis. At the core of our model, we propose an energy-efficient algorithm by means of which individual sensors are able to transmit data over shortest path. In simulated experiments, results reveal that our method yields a stable packet delivery ratio. It is also observed to be more energy efficient compared to models that use cluster head selection in wireless sensor networks. Using this model, a region in a forest or wildlife habitat can be monitored for an extended period without compromising network throughput.

Keywords: Cloud computing, Markov chain, Smart forest, Wireless sensor network (WSN)

Background

Introduction

There is a growing awareness to save and conserve wildlife so that future generations can enjoy the biodiversity of our planet. As it is, we are witnessing changes in climate around various regions around the globe (Yavasli et al. 2015). These climatic changes have caused natural habitats that have existed for aeons to slowly disappear. Without a suitable environment in which to survive, many species have begun to go extinct (Allsopp et al. 2012). Around the world, researchers are concentrating on ways to protect not wildlife, but also the flora and fauna that support the existence of native wildlife. One of the most debilitating factors threatening conservation efforts are incidents of forest fires. The United States Forest Service and other similar agencies take this threat very seriously. Forest fires not only cause extensive damage in areas where these occur, but also influence weather patterns. With the help of applications developed based on Internet of Everything (Bradley et al. 2013) and Internet of Things (Biet 2014), increase in carbon monoxide levels can be detected in advance. Using data generated, researchers can provide visual cues to residents living in and around forested areas and to the public at large.

The case for automation and wireless sensors

For reasons related to health and safety, humans cannot be deployed in forest areas and especially in dry seasons where forest fires can break out without warning. Recording stations and posts are also somewhat limited when it comes to adaptability. Wireless sensor network technology is comparatively cheaper and more readily deployable. Further, the technology can be automated to collect information not only of the environment, but also of wildlife in motion. For these reasons alone, wireless sensor network technology can be a boon for researchers and agencies engaged in preservation and monitoring of wildlife in forests (Trifa et al. 2007).

The challenges posed in operating in such conditions are significant. Camera traps are useful and provide important visual information. However, these cannot be deployed in significant numbers without incurring considerable costs. Cameras also required batteries to operate. While motion sensors can set off a camera for recording either still or video images, power consumption is quite high and, therefore, require frequent replenishments in the form of fresh batteries. Wireless sensors, on the other hand, have lower power requirements and can remain in the field for a longer period of time (Garcia-Sanchez et al. 2010).

For any detection and monitoring system to be effective, it has to be relatively cheap and has to have a long life in practical conditions (Dyo et al. 2010). Therefore, the objective of implementing any wireless sensor technology is to be practical, reliable and self-aware. In combination with radio tracking collars and static sensors, WSN technology can help track movements of animals and continually monitor local weather and climatic conditions.

Network life and network throughput: maximizing value on investment

Reducing maintenance costs involved in replacing batteries for sensors yields a longer life for the overall network. In our paper, we approach this aspect with the ultimate aim of sustaining network life and network throughput. We do so through a mix of deploying fixed and mobile sensors in a pre-determined ratio. This ratio is dependent on the application and eventual deployment conditions. We have designed a model in which information is relayed through a path which involves the least number of mobile sensors. We also take into account the possibility of premature death of a certain percentage of sensors. The technique we propose relies on this assumption and selects the best route suitable for sending information packet to the selected destination. Reducing the number of sensors involved for transmission, in turn, minimizes overall network power consumption. As a result, life of the network can be extended and a higher network duty cycle can be achieved. Consequently, researchers and agencies engaged in wildlife conservation can sift through larger amounts of data while keeping maintenance and investments under greater control.

American biologist and explorer, Sylvia Earle, once remarked that wildlife was more abundant than the population of almost two billion human beings when she was born. The situation, she observed, is now reversed. While many others have tried to impress upon the need to conserve forests and protect wildlife in their natural environments, few have encapsulated this thinking more succinctly the way she has.

Related work

In a preliminary study conducted by EFR "Smart Forests" Working Group (Rustad and Stine 2013) on the cyber infrastructure capacity and needs at USDA Forest Service Experimental Forests and Ranges for the twentyfirst century, the significance of wireless sensor networks to monitor environment at forest reserves or wildlife parks has been discussed. The primary objective of this study was to obtain an almost instantaneous access to precise sensor data from several remote locations and coalesce these into a single website where these can then be studied. The report also determined that at the time the results of this study were obtained, 28 % of all EFR sites across the United States had been actively considering putting in place wireless sensor networks to monitor the environment. The study further determines data transmission to be the most widely occurring impediment in setting up a competent wireless sensor network. Several authors (Rustad et al. 2014) have used this study as the basis of their combined research efforts in harnessing the usefulness and efficacy of wireless sensor networks to obtain real-time information on meteorology and other environmental factors.

The significance and importance of monitoring wildlife environment and habitats in the India has been recognized by successive governments in power at New Delhi. The Constitution of India has specific provisions and directive declaring state policy (The Consititution of India, 1st December, 2007) which empowers the Ministry of Environment, Forest and Climate Change, Government of India to protect and improve forest and wildlife regions across the country. Studies conducted in the United States and measures adopted are closely observed for possible adoption in the context of several wildlife parks and protected forests. The United States Department of Agriculture (USDA) Forest Service has implemented several projects over the years using digital sensors that wirelessly communicate with central servers in order to actively analyze various environmental parameters. A substantial portion of the technology used has been adopted in India by the concerned authorities to protect hundreds of wildlife species that include animals and wide variety of trees and plants.

Forest fires can and often leave a devastating impact on ecosystems. Extremely large forest fires can cause widespread loss and damage not only in terms of environment but also insofar as commercial interests are concerned. Lyon et al. (2000) have discussed fire regimes and the manner in which these shape and change the landscape (Lyon et al. 2000). In their work in designing a Fire-smart Forest Management approach, Hirsch et al. (2001) have discussed the challenges involved in the context of Canada' forest regions. The authors identify the need to minimize the economic and social impacts left by devastating forest fires over the most widespread and commonly-held belief that forest fires should be controlled in an efficient manner. Designing a reliable method of detecting forest fires has been the core focus of the work of Bouabdellah et al. (2013). The authors discuss wireless sensor networks (Bouabdellah et al. 2013) in the context of forest fires, and compare two different fire detection methods of Korean and Canadian origin. The study reveals that the Canadian method is significantly more accurate than that of the Korean one. Presenting a design for an early warning and fire detection system using wireless sensor technology, Hefeeda and Bagheri (2008) analyze Fire Weather Index (FWI) system prevalent in North America. The authors propose algorithms (Hefeeda and Bagheri 2008) which are aimed at extending network lifetime of the sensors deployed as well as provide an effective coverage umbrella across different zones in the forest. Based on ZigBee Wireless sensor network, Zhu et al. (2012) have presented a fire detection system (Zhu et al. 2012) that conveys data in real-time.

Flood detection

An equally concerning natural disaster are incidents of floods. Floods can leave devastating impact on wildlife and ecosystems. In Gujarat, a western state in India, flash floods have left ten lions, over a thousand blue bulls, and around ninety spotted deer killed in 2015. This incident, and similar ones which have happened in the past, have attracted the attention of experts to focus on flood detection research using sensor network technologies. Pasi and Bhave (2015), present a unique method involving underwater and surface sensors that actively monitor the flow of water current. The rate of flow is analyzed in real-time to determine if a threat exists as a result of an evolving flood situation.

Basha and Rus (2007), propose an autonomous sensing system comprising two communication tiers of radio networks. The authors examine and test the possibility of using a combination of medium to long range radio links (Basha and Rus 2007) replacing wireless sensors in areas where recurrent sensor node losses may not be acceptable for communities in remote regions. Existing radio communications can be used in conjunction with underwater sensors protected using materials that can be sourced cheaply and locally.

Quantitative analysis of water flow patterns in rivers is presented in the work of Kugker and De Groeve (2007). A flood trigger is enabled on the basis of histogram of time series data spread over a period of 4 years. Crossing of a threshold mark of 80 % of the cumulative frequency of this histogram would automatically trigger a flood alarm (Kugker and De Groeve 2007). While this research largely involves use of data obtained through satellite technology, the method proposed by Lo et al. (2015) is an interesting alternative (Lo et al. 2015) to the previous mentioned work. Building on surveillance systems and image processing methods, Lo et al. present a system which acts as an intrusion detection device where a developing flood situation is deemed as a possible invasive object.

Early-warning: earthquakes and tsunamis

Use of sensors and sensor-based technology has found acceptance in the work of Allen (2011). His work is directed at detecting earthquakes as early as possible using a system of sensors that transmit data to a central site (Allen 2011). The work of Tan et al. (2013) explores wireless sensor technology to detect seismic occurrences. The authors propose a dynamic sensor selection (Tan et al. 2013) algorithm to minimize incidents of false alarms and increase sensor detection rates. Designing a system that senses and responds to tsunamis has been the focus of Casey et al. (2008). This work uses an ad hoc sensor network and route repair algorithm to detect and repair failures in data transmission.

Characteristic feature: wireless sensor networks

As a means to monitor an environment and gather information on the changes taking place within the confines of such, a wireless sensor network helps fulfill a vital role. A wireless sensor network is an array of sensor nodes that interact with each other to gather data and transmit the information to a central processing center. Wireless sensor networks employ hops to send data packets from one node to the next. The final destination in most cases is reached via satellite or Internet, or using a combination of both. Several routing protocols and topologies are used in order to maximize efficiency in wireless sensor networks. Flat routing, hierarchical routing, multi-path routing, query-based routing, QoS based routing, and various types of associated network topologies are well known. However most of these suffer from lack of adequate battery power sufficient to sustain prolonged processing power needs of sensor nodes. As a result, several energy-aware algorithms have been proposed which rely primarily on cluster-based methods.

In 2000, Heinzelman et al. (2002) proposed an approach which implemented a hierarchical routing protocol using clusters. The proposed method was based on LEACH, wherein the details of the nodes are handled by cluster heads. It is the job of cluster heads to collect and compress data before dispatching to the respective sink nodes. The advantage of such an algorithm is that it is scalable and is an efficient method to transmit data between cluster heads and sensor nodes. However, this approach is not without its disadvantage. Formation of clusters in each round is neither energy-efficient nor does it support sensor nodes which have to be mobile. Kim and Chung (2006) proposed LEACH-Mobile (LEACH-M) routing protocol which was introduced as a way to improve existing LEACH-based protocols. LEACH-M focuses on decision-making of cluster and confirms status of communication of mobile sensor nodes with a specific cluster head. But this, too, imposes a significant overhead as messages are initiated to declare membership.

To alleviate the problem, Kumar et al. (2008) proposed LEACH-Mobile Enhance (LEACH-ME). The determinant in this proposed method is mobility which influences the selection of cluster head. However, this algorithm also consumes more energy so as to be able to identify mobility factors for nodes in the network. Anitha and Kamalakkannan (2013) presented an advancement over LEACH-M protocol. In their paper, the authors presented EEDBC-M. The unique feature of this design is the manner in which cluster is formed as a result of computation of factors involving mobility, residual energy and K-density measures. In 2012, Zhu and Jia (2012) presented a novel hybrid self-organizing cluster routing algorithm. The unique feature of this design is the incorporation of self-organizing behavior in the nodes to identify the on-demand route data packs from source to sinks based on residual energy of an elected cluster head and stability of connection.

Through our literature survey and assessment of the existing works published by the respective authors, we have identified the need for a new on-demand routing protocol which can support self-organizing behavior of nodes while minimizing costs towards cluster head selection procedures. We present a self-servicing routing protocol that is able to do away with energy consumption required for selection of cluster heads without compromising loss of data during transmissions. Our proposed design focuses on establishing and maintaining connectedness without involving cluster heads.

Fusing static and mobile wireless sensor networks

In his dissertation on tracking wildlife, Markham (2008), discusses the efficacy of wireless sensor technology in the context of wild animals in their natural habitat. Installing wired medium or physically collecting data from nodes embedded deep in the forests can pose difficulties and challenges. The introduction of wireless sensor networks (Fraunhofer-Gesellschaft, "Research News-Fraunhofer" 2011) makes it significantly easier to funnel data to a downloading station through one or more devices which act as intermediaries. Inclusion of wireless network elevates tracking solutions from being just animal-monitoring approach only to a broader scope involving forest and wildlife management. Data collected from low-end devices can be collected by instruments with a wider range of functionalities. These can then transmit the information to the users concerned. The concept introduced in this paper is an amalgam of stationary and mobile networks. To be effective and relevant, the system has to be able to convey information to the user via stationary and partially mobile nodes at the same time.

Power sources for sensor nodes

The subject of electrical sources for powering sensor nodes is a prominent topic in this area of research (Shnayder et al. 2004). In an often hostile and remote environment, sensor nodes are often deployed with their own sources of electrical power—usually in the form of batteries. While the task of replacing batteries alone can be an arduous, a stable functioning of the wireless sensor network can be easily disrupted in the process. This translates into losing of data packets, which, in turn, causes the quality and accuracy of the system as a whole to go downhill. In theory and in practice, one of the chief methods employed to prolong service life of a wireless sensor node has been to cause the onboard radio to go dormant. It has been observed that while energy consumed is about the same in receiving, transmission, or idle modes, it is the sleep state which consumes the least amount of power in comparison to all other modes. The goal, therefore, is to reduce the number of hops it takes to transmit data from a wireless sensor to the central collection point. This leads to a design where each wireless sensor works with its nearest sensors to push their data packets to the intended destination through a path which offers the optimum energy consumption. Prelude to such self-servicing design can be evidenced in the works (Fraunhofer-Gesellschaft, "Research News-Fraunhofer" 2011; Markham 2008) discussed. Although, advancements are continually being made in designing batteries which last longer (Fraunhofer Institute for Industrial Mathematics 2015), the newer designs remain yet to be tested (Shnayder et al. 2004; Lorincz et al. 2004). In the wake of such promising developments, researchers continue to examine possible solutions that can make wireless sensor networks more appropriate and efficient in harsh and otherwise inhospitable settings. Wireless sensor networks offer more flexibility and ease of deployment that wired networks. But with this advantage comes the cost of replacing or replenishing power sources of the nodes or network motes.

The cloud connection

One of our underlying objectives has been to make our proposed design collect raw data and stream the information back to a cloud-based system. We realized that the efficacy of our design can be enhanced by introduction of an efficient and environment-friendly cloud-based solution. A mobile cloud computing approach as discussed in the work of Sarddar et al. (2015) was examined (Sarddar et al. 2015) and adopted in our work.

Problem formulation

To build a wireless sensor network that is effective and quick to record changes in an environmental setting poses a significant challenge. While the shortest path would generally represent the most effective way to transmit data from a particular node to the sink node or the central storage point, this could also involve quite a large number of packet drops leading to decline in quality of transmission of relevant information.

It becomes, therefore, imperative that a design be conceived to minimize data loss whilst improving fault tolerance. Not only does this improve quality of data transmission, it also is a step forward in addressing a perennial question of conserving energy to the extent possible.

Determining the optimum path

To compute the best path under a set of circumstances, we formulate a path score. The path score is an indication of the relevance and best-fit path among a set of paths available for data to be transmitted from a given node to its target destination. We represent the set of total available paths from a node to its sink node as *Path*. Thus, *Path* is defined as follows:

$$Path = \left\{Path_i\right\}, \quad \text{where } 1 \leq i \leq k, \quad \text{k} = \text{Total number of paths.}$$

where, k represents the total number of paths; and i represents each path among the set of paths (we use the terms "sink node" and "central collection point" interchangeably from this point onwards).

Representing the number of links in a given path

A given path has a number of links joining one sensor to another to form a single continuous chain. We define L_{ij} as the link between two sensor nodes; say N_i and N_j, in a given path. Thus, for the total number of links, m, in $Path_i$, L_{ij} is defined as {(Ni, Nj)} where $1 \leq i, j \leq m$.

The two-state Markov chain

Whether a link between two nodes can be established successfully is determined by a two-state Markov chain. The two-state Markov chain determines whether a state of acceptance can be achieved considering the level of energy of the recipient node.

Link availability is shown in the transition matrix presented in Fig. 1.

This transition matrix is applied to the two-state Markov chain as shown in Fig. 2.

The two-state Markov chain depicts a state of acceptance as A, and a state of rejection to be R. Further, the probabilities represented by the transition matrix in Fig. 1, are explained as follows:

$$P = \begin{pmatrix} S_1 & S_2 \\ S_3 & S_4 \end{pmatrix}$$

Fig. 1 Transition matrix

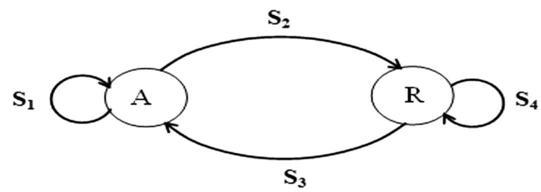

Fig. 2 Two-state Markov chain

S_1: Probability that the next state would be A given that present state is also A.
S_2: Probability that the next state would be R given that present state is A.
S_3: Probability that the next state would be A given that present state is R.
S_4: Probability that the next state would be R given that present state is also R.

The sum of each row in the transition matrix is 1.

Node stability

Stability of each node is computed in order to determine whether a stable path can be established. Node stability is determined by the sum of three distinct values involving (a) strength of available energy resource of the node; (b) mobility index of node; and (c) chances of premature death of node.

Weighted priority of each of these is declared to be w_1, w_2, and w_3, respectively, where:

$$w_1 > w_2 > w_3 \text{ and } w_1 + w_2 + w_3 = 1$$

The equation for arriving at the *Node Stability* value for a given node is given as follows:

$$node\_stability(N) = P + Q + D \tag{1}$$

where, P is given as $P = w_1 \times probability(accept\_state) = w_1 \times Maximum(S_1, S_3)$. Q is given as $Q = (-w_2) \times probability\left(mobility\_chance\right) = (-w_2) \times \frac{V_P}{V_{MAX}}$. And, D is computed using $D = (-w_3) \times probability(premature\_death)$.

To arrive at the value of P, we multiply the weighted priority index to the maximum of the states S_1 and S_3.

The value of Q is taken to be negative as more mobile a node is, lesser are the chances of producing a stable linked node. While V_P represents the current velocity of the node, V_{Max} indicates the maximum velocity that the node can achieve. In physical terms, this factor assumes a great deal of significance when viewed from the point of a node strapped to a wild animal.

The last parameter D is determined by the probability of a node suffering a premature death. The values of Q and D are treated as negative values for two reasons. A node attached to a wild animal approaching its maximum velocity would tend to lose its efficacy in being able to effectively receive and transmit data. With a high probability of premature death, a target node for communicating data is likely to cause errors and transmission failures.

We use Poisson Probability Distribution to calculate the chance of a premature of a sensor node. The probability mass function is given as follows:

$$P(X = k) = \frac{e^{-\mu}\mu^k}{k!}$$

where k = 0, 1, 2, 3… and μ is the expected value of X.

Establishing the relationship between link and node stabilities

Thus, link stability is computed as follows:

$$link\_stability(N_i, N_j) = \frac{node\_stability(N_i) + node\_stability(N_j)}{2} \quad (2)$$

Evaluating path score of a path

We proceed to count the total number of mobile nodes in a given path. This is given as follows:

$$L_{score}(Path_i) = count(NM) \quad \text{where NM represents mobile nodes} \in Path_i$$

Path stability (*path_stability*) of each path can, therefore, be defined by $PP(Path_i)$. This value is evaluated using the following function:

$$PP(Path_i) = \prod_{L_k \in (Path_j)} probability(L_k)$$

Path score can now be evaluated by maximizing the following objective function:

$$f(.) = c_1 \times \left(\frac{Path_i\_stability}{\max(Path\_stability)} \right) + c_2 \times \left(\frac{hop\_count_i}{\max(hop\_count)} \right) + c_3 \times \left(\frac{L_{Score}}{\max(L_{Score})} \right) \quad (3)$$

where, $|c_1| + |c_2| + |c_3| = 1$, and $c_1 \rangle c_2 \rangle c_3$.

Evaluating the highest path score to determine the best path

Once all the path scores have been computed, the maximum score is determined. This score gives the path which offers the maximum stability for transmission of data to occur between a node to its corresponding sink node.

$$Path\_score = \max_{\forall Path_i} f(Path_i) \quad (4)$$

Proposed methodology

The objective behind our proposed routing strategy is to actively ascertain the most appropriate route in a wireless sensor network deployed in any forest or remote location.

Primarily developed for providing a reliable communication path from source to destination, wireless sensor networks were designed with designated routes in order to avoid unstable connections. We propose a self-servicing energy efficient routing strategy that is aimed primarily to be deployed in locations where survival and well-being of heritage and rare animals are of utmost importance. In composing our work and formulating our approach, we have envisaged a scenario where two types of wireless sensor nodes are used. Those which are embedded in trees or mounted on any other fixed structure are called fixed sensors. Attached to neck collars of tagged wild animals, the second type of wireless sensors is called mobile wireless sensor (Markham 2008). A conceptual representation of a sample area covered by our proposed wireless sensor network design is presented in Fig. 3.

Our proposed methodology is composed of three distinct sections that together combine to function as a single design paradigm. These are as follows:

1. Friend Sensor Node Discovery.
2. Stable Route Establishment.
3. Route Maintenance.

Our proposed algorithm itself has its roots in the following assumptions:

1. Each sensor node is instantiated with an adequate amount of energy for operations.
2. Each sensor node has its own locational information embedded in it.
3. The wireless sensor node network is self-contained with no entry points for malicious programs to infiltrate through into the system.
4. Each node is able to transmit packets of data to all other sensor nodes that are friendly.
5. Three variables are constantly maintained and monitored by each sensor node. These variables are named RESIDUAL_ENERGY, MOBILITY_CHANCES, and PREMA-

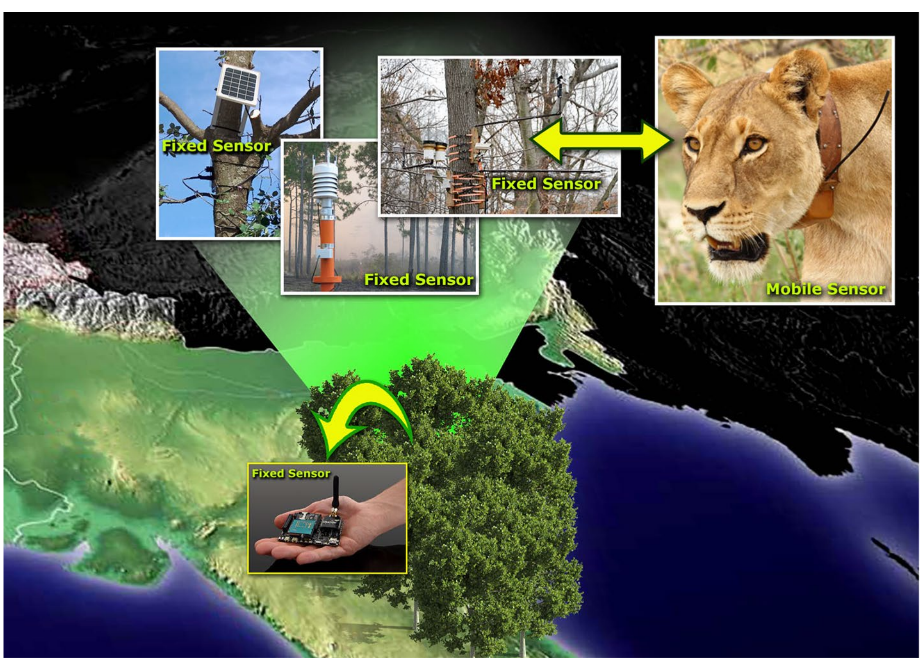

Fig. 3 Conceptual view of placed sensor over an area

TURE_DEATH. The eponymous names refer to the energy remaining in a sensor node, chances of mobility associated with the sensor node, and life expectancy of each node measured by Poisson Probability Distribution, respectively. It is interesting to note that in our work, we have held the variable MOBILITY_CHANCES along with PREMATURE_DEATH to be as negative values to be considered in our calculations for evaluation of link and node stabilities. The reason being, higher the chances of a node being mobile, lesser would be the quality of transmission of data packets involved. Hence, a sensor node that is attached to a tracking collar of a wild beast may either be at rest or in motion. Depending on the mobility factor, evaluation of node stability is made.

6. An array is independently maintained by each sensor. This array constitutes information on link stabilities with corresponding friend sensor nodes.
7. Each sensor node maintains information on its active friend sensor nodes.
8. Only friend sensor nodes are able to receive packets from other active corresponding sensor nodes.

We present a pictorial representation of our proposed wireless sensor network in Fig. 4.

Friend sensor node discovery

In order to receive information from its neighboring nodes which are one hop away, a node broadcasts an ENQUIRY packet to determine its nearest neighbor within its range.

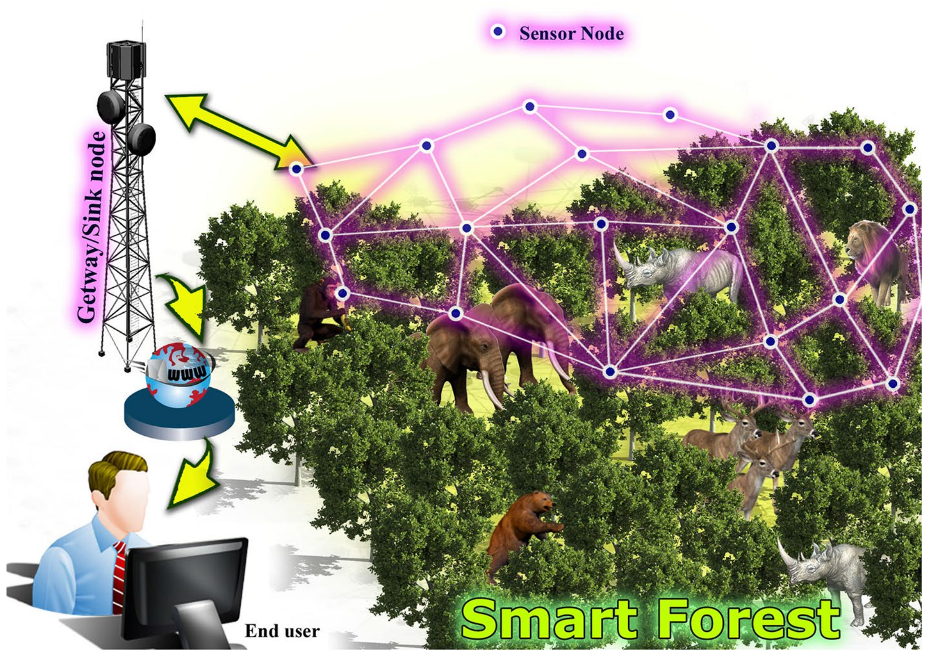

Fig. 4 Our proposed WSN architecture for smart forest

On receiving one such ENQUIRY packet, a node acknowledges receipt by sending a return ENQUIRY_ACK packet. This packet of information contains identity data of the node and the node stability value computed through equation shown in the Problem Formulation section. Only after receiving node stability values, the initiating node computes ideal route through a process described in the following section.

Stable route establishment

To be able to establish a stable route to the destination, a source node has to have corresponding ENQUIRY_ACK packets from its friend sensor nodes. A source node then calculates link stability using equation detailed in the Problem Formulation section. A node continually seeks out the best link among all friend sensor nodes. However, link stability is of importance, and needs to be in the range of values between 0.35 and 0.45. A ROUTE_REQUEST (RREQ) packet is sent to a friend sensor node only if the link stability value falls within this range and is the highest among others. If the link stability values fall in the range of 0.15–0.34, then ROUTE_REQUEST packets are sent to two nodes which have values higher than the remaining ones. In any other event, ROUTE_REQUEST packets are sent to all friendly sensor nodes. After receiving ROUTE_REQUEST packet, the corresponding sensor forwards it to the sink node using the method described above. To prevent errors arising out of loops, intermediate sensor node appends its own ID along with that of the sending node in the ROUTE_REQUEST packet. When such ROUTE_REQUEST packets reach the sink node through different paths, the sink node then begins to calculate the maximum path score using the equation presented in the Problem Formulation section. Upon determining the highest path score value, the sink node sends a ROUTE_REPLY (RREP) packet only to the initiating source node. The source node confirms the path to the sink node following receipt of the ROUTE_REPLY packet. This path is used eventually to transmit data back to the sink node.

Route maintenance

A route cache is maintained by each and every sensor node in the network. This cache is updated at specific intervals to facilitate in performing quick calculations at high speeds.

Proposed algorithm

Algorithm of our proposed Self-Servicing Energy Efficient Routing Strategy (SSEER) is discussed below:

SSEER ()
Input: Considered Network Topology, Source, Destination
Output: Stable path from source to destination
1. Begin
2. existPath ← Call CheckRouteCache ()
3. if (existRoute != NULL)
3.1 Sent the data packet along the path enlisted in route cache
4. else
4.1 Call FriendSensorNodeDiscovery ()
4.2 Call RouteEstablish ()
4.3 Call RouteMaintenance ()
5. End

CheckRouteCache ()
Input: Source, Destination, Timestamp, Route Cache Table
// each row of Route Cache table consists of the entry i.e. <Node id, Source, Destination, Stable path, Current Timestamp>
1. Begin
2. existRoute ← NULL
3. CurrentTimestamp ← Timestamp
3. if (Row.Source = Source && Row.destination = Destination && (CurrentTimestamp − Row.Timestamp) ≤ ThresholdTimestamp)
3.1 Path ← Stable path
3.2 Return Path
4. End

FriendSensorNodeDiscovery ()
Input: Source, Destination
Output: Generates a set of friendly sensor nodes
1. Begin
2. currentNode ← Source
3. while (currentNode != Destination)
3.1 Broadcast ENQUIRY packet to its friend sensor nodes s.
// ENQUIRY packet consists of sender's node id.
3.2 On receiving ENQUIRY packet each receiving sensor node checks the residual energy of the receiving sensor node.
// Set w1 = 0.5, w2 = −0.3 and w3 = −0.2
3.3 if RESIDUAL_ENERGY > 75% of TOTAL_ENERGY
3.3.1 S1 = S3 = 0.9, computes node stability as shown in the Problem Formulation section.
3.4 else if RESIDUAL_ENERGY ∈ [25%, 74%] of TOTAL_ENERGY
3.4.1 S1 = S3 = 0.7, computes node stability using as shown in the Problem Formulation section.
3.5 else
3.5.1 S1 = S3 = 0.3, computes node stability using as shown in the Problem Formulation section.
3.6 End if
3.7 Each receiver sensor node sends ENQUIRY_ACK packet to the sender node id along with its node stability.
// ENQUIRY_ACK packet consists of node stability of that particular node (fn).
3.8 After receiving each ENQUIRY_ACK packet, each sensor node s does Enqueue (fn, s)
3.9 Generates an adjacency list of friendly sensor nodes for each current sensor node i.e. {fni}currentNode, i = 1 to total number of friendly sensor nodes
3.10 currentNode ← Dequeue (fn, s)
4. End while
5. End

RouteEstablish ()
Input: Source, Destination, Set friendly sensor node for each of the nodes from source to destination
Output: Stable path from source to destination
1. Begin
2. currentNode ← Source
3. while (currentNode != Destination)
3.1 Computes link stability of each friendly sensor node of currentNode as shown in the Problem Formulation section.
3.2 Evaluates the maximum link stability among neighborNode (currentNode)
4. if (max (link_stability (neighborNode (currentNode))) ∈ [0.35, 0.45])
4.1 selectedNode ← {node}, where node is the neighborNode corresponding to the highest link stability.
4.2 else if (max (link_stability (neighborNode (currentNode))) ∈ [0.15, 0.34])
4.2.1 selectedNode ← {node1, node2}, where node1 and node2 are the neighbor nodes corresponding to the highest link stability and second highest link stability respectively.
4.3. else
4.3.1 selectedNode ← {fni} i.e. set of all friendly sensor nodes.
5. End if
6. Enqueue (node), ∀ node ∈ selectedNode.

7. currentNode sends RREQ packet to each node of selectedNode.
//RREQ packet includes the following entries:
<currentNode, selectedNode, Path (Source, selectedNode), path_stabiliy (Source, selectedNode)>
// Set c1 = 0.5, c2 = −0.3, c3 = −0.2
// path stability from Source to selectedNode is computed as shown in the Problem Formulation section.
8. currentNode ← Dequeue (node), ∀ node ∈ selectedNode.
9. End while
10. Destination node enlists the all Path to the routing table along with current timestamp from all received RREQ packets; now path value is computed as shown in the Problem Formulation section.
11. For finding the stable path, computes the path score as shown in the Problem Formulation section for all paths.
12. Destination node now sends RREP packet to the source node through the stable path.
13. Source sensor node sends the data packet along the discovered stable path and wait for threshold time stamp.
14. Send data acknowledgement packet to the source node.
15. If (time stamp is exceeded and data acknowledgement packet is not received)
15.1 Delete the all entries of the selected row from the route cache
15.2 Update route cache
16. End if

RouteMaintenance ()
Begin
1. if (CurrentTimestamp – Row.Timestamp ≥ ThresholdTimestamp)
1.1 Delete the all entries of the selected row from the route cache
1.2 Update route cache
2. End if
3. End

Example of our proposed methodology

We illustrate with an example the working of our proposed methodology. This is presented in a graphical format as show in Fig. 5. A circle represents each node and its corresponding node ID. An edge denotes individual link between two nodes. A link drawn connecting nodes indicates that the nodes exist within mutual range of transmission. As discussed in "Problem Formulation" section above and tabulated in Table 1, each of the nodes maintains three variables. The node stability of each is calculated in the "Problem Formulation" section and is exemplified in Table 1.

Table 2 comprises two sections. The first represents the neighboring node IDs of a given node (represented by its own node ID). The second section tabulates the link stability values between two nodes. The calculation of which has been explained in the Problem Formulation section. The RREQ packet delivery mode is clarified in Fig. 6a–g.

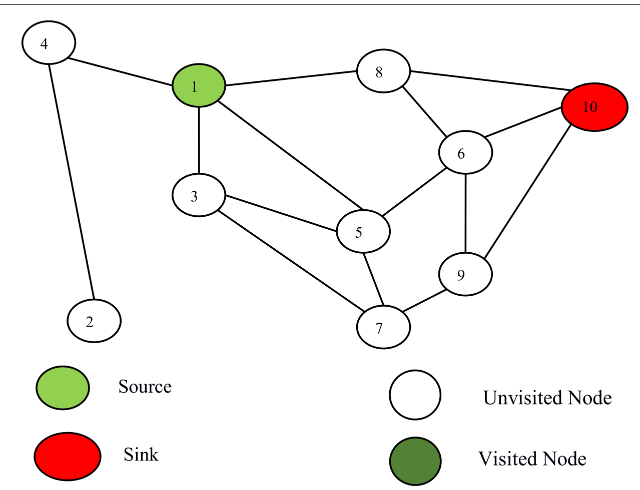

Fig. 5 Our proposed WSN for smart forest

Table 1 Stability of each sensor node along with ID

Node Id	Residual energy (%)	Activeness	Mobility	Sensor node stability
1	90	0.9	0.001	0.4317
2	85	0.9	0.7	0.222
3	78	0.9	0.25	0.357
4	70	0.7	0.02	0.326
5	68	0.7	0.02	0.326
6	85	0.9	0.25	0.357
7	50	0.7	0.03	0.323
8	60	0.7	0.04	0.320
9	78	0.9	0.01	0.429
10	95	0.9	0.02	0.426

Table 2 Link stability along with node ID

Node Id	Neighbor nodes			
1	3	4	5	8
2	4			
3	1	5	7	
4	1	2		
5	1	3	6	7
6	5	8	9	10
7	3	5	9	
8	1	6	10	
9	6	7	10	
10	6	8	9	
Link				**Link stability**
(1, 3)				0.39
(1, 4)				0.37
(1, 5)				0.37
(1, 8)				0.37
(2, 4)				0.27
(3, 5)				0.34
(3, 7)				0.34
(5, 6)				0.34
(5, 7)				0.32
(6, 8)				0.33
(6, 9)				0.39
(6, 10)				0.39
(7, 9)				0.37
(8, 10)				0.37
(9, 10)				0.42

In the example shown, we assume that node ID #1 wants to send data to a sink node which has a node ID #10. We assume that each node receives ENQUIRY_ACK packets from its corresponding friendly sensor nodes. Every ENQUIRY_ACK packet which is transmitted contains a node stability value of the issuing node. As seen in the example,

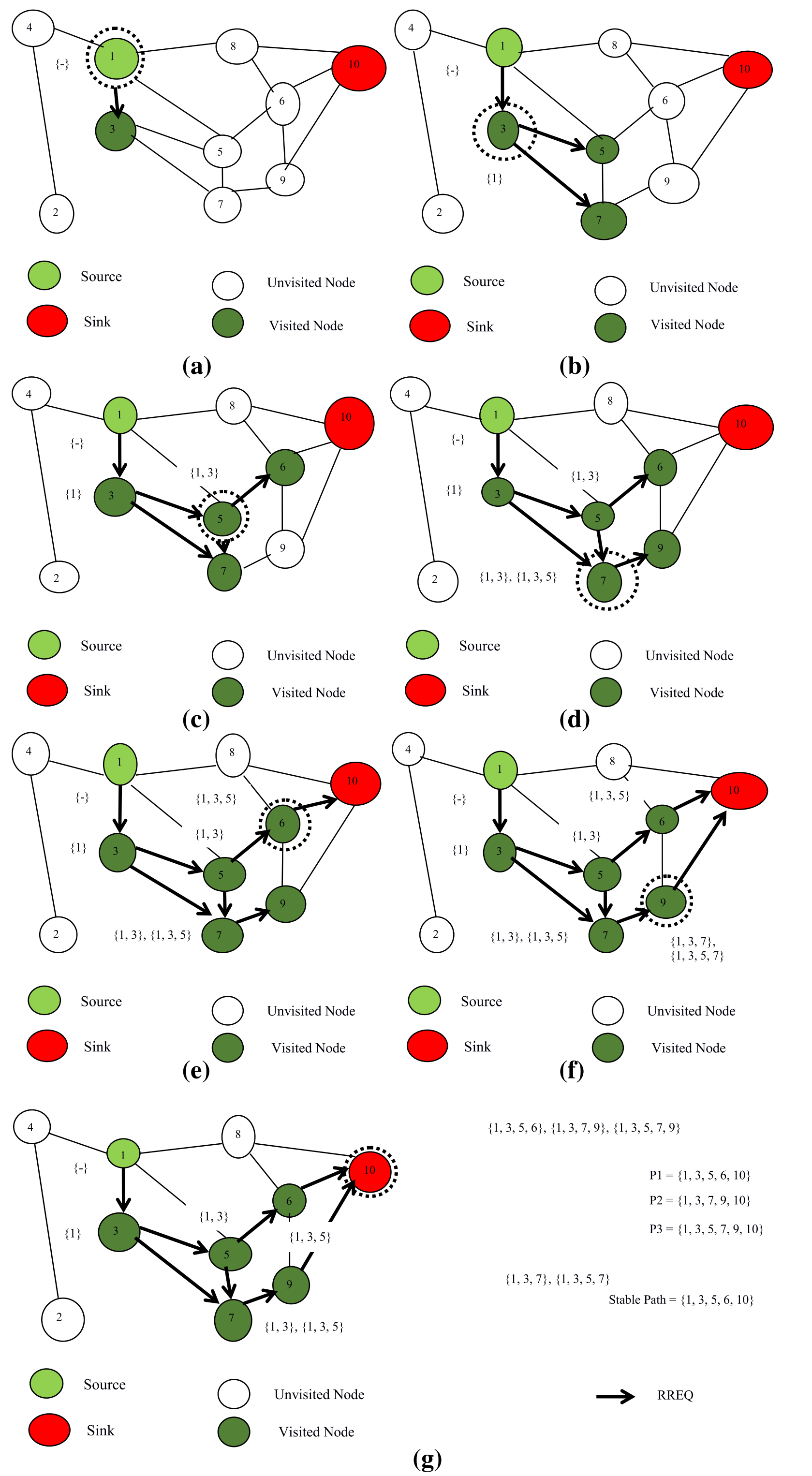

Fig. 6 a–**g** Step wise graphical representation of RREQ packet delivery using our proposed methodology node ID #1 determines that the maximum link stability is accorded by node ID #3 with a value of 0.39. Hence, node ID #1 sends a RREQ packet to node ID #3 which is determined to be the healthiest. In a similar manner, node ID #3 computes link stabilities of its neighboring nodes. It finds node ID #5 to be healthiest with a value of link_stability is 0.34. This process goes on till the RREQ packet reaches sink node ID #10. Upon receipt of this packet, the sink node is able to discern the most stable path after sorting the path score values in descending order with the highest score at the top. A RREP packet is then transmitted back to the initiating node through the stable path identified resulting out of this process. The transmission is unicast and does not affect nodes which are not part of the path determined to be as the most stable for the transmission. The source node begins transferring data over this stable path to the sink node (Fig. 7a–b).

At this stage, it is important to discuss the possibility of premature death of a sensor. We assume that such a chance is at 0.01. Let us also assume that X denotes the number of nodes which have suffered a premature death. We may then say that the probability that a sample size of 10 sensor nodes having exactly 10 % nodes that have died prematurely, is calculated by $P(X = 1) = e^{0.1} \times (0.1)^1 = 0.09$.

Experimental set-up and simulation analysis

We have proposed a design wherein a mix of static and dynamic mobile sensor nodes have been used. Static sensors are deployed on trees or affixed to towers located at various vantage points in a forest. Animals which are on the endangered species list, or those identified as heritage animals, have collars attached with mobile sensors. The purpose of the static sensors is to collect various metrics related to weather and local environment patterns. Mobile sensors collect data related to the general well-being of the animal. Such data includes body temperature, and speed of movement of a given animal.

To demonstrate the workings of our proposed method, we have considered a deployment of almost 100 static and 50 mobile nodes randomly deployed over an area 250,000 square meters. In our experiment, the maximum range of communication per sensor was set at 100 meters. All sensor nodes had their power replenished through solar panels which eliminated the need to visit each node in order to replace drained-out batteries.

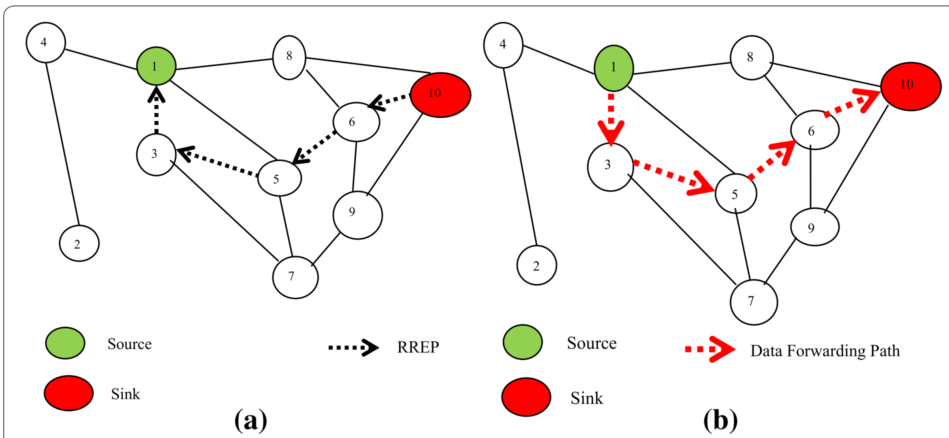

Fig. 7 a, **b** RREP and data packet transmission

Within their respective range of transmissions, each node could freely talk to another. We used 10 J as initial energy for each sensor. In order to transmit and receive signals, a MICA2 energy model was adopted (Shnayder et al. 2004; Lorincz et al. 2004). It was calculated that approximately 2.34 and 4.602 μJ/bit is required for receiving and transmitting signals, respectively, for data packets, ENQUIRY packet, ENQUIRY_ACK, ROUTE_REQUEST and ROUTE_REPLY are considered to be sized 512, 5, 5, 16 and 16 bytes respectively.

In order to achieve ideal simulation results, we varied the sensor node density from 100 to 280. In the course of our simulated studies, the average mobility was kept at 0.5. Premature death of up to 5 % in a round was assumed to be 0.4. Our simulation was conducted over 10–60 rounds in a random network setting.

In the emulation experiment, we select LEACH-M and LEACH-ME algorithm as references and do simulation experiments to the proposed algorithm via LEACH-M and LEACH-ME (Kim and Chung 2006; Kumar et al. 2008), and select some representative simulation results to be analyzed and discussed.

Packet delivery ratio

It was important for us to determine the packet delivery ratio to understand whether the desired results were being achieved by our proposed model. A packet delivery ratio (PDR) is computed as a ratio of the total number packets sent by a node to the total number of packets received at the destination. The formula for this is presented below.

$$PDR = \left(\frac{R}{S}\right) \times 100 \quad (5)$$

R = Total number of packets send by the source node; S = Total number of packets received by the destination node.

As can be seen in Fig. 8, the packet delivery ratio in our experiment was observed to be better PDR value with the reference algorithms LEACH-M and LEACH-ME in relation

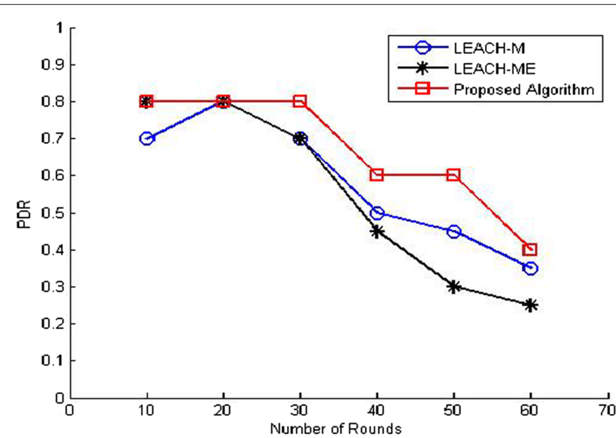

Fig. 8 Packet delivery ratio versus number of rounds

to number of rounds because in these two methods elected cluster head die before the end of the round and node in this cluster loses their data.

Average energy consumption rate

Energy consumption being of significance, it was important to determine how this varied in relation to number of nodes. The formula for arriving at an average energy consumption rate is given as follows:

$$Average\ Energy\ Consumption\ Rate = \left(\frac{E_I - E_C}{E_I}\right) \times 100 \tag{6}$$

where E_I represents initial energy of a node; and E_C denotes energy consumed after n number of rounds.

It is observed from the graph as seen in Fig. 9, that though the average rate of energy consumption rise in proportion with increasing number of nodes, it offers improvement over the other two methods because no energy consumption is requirement for cluster head election.

Conclusion

In an era where large chunks of data needs to be processed in real-time, cloud computing offers the advantage of allowing access from a common repository from any number of location. The ability to constantly pool data into a central cloud storage system by a self-servicing wireless sensor network offers a significant advantage to forest and wildlife conservationists and experts. Analysts and researchers can study and design better solutions that can be agile and robust in changing environments. Further, loss of life and wildlife habitat can be significantly reduced by means of advanced early warning feature inherent to our proposed system.

Each sensor node is equipped with the ability of selecting an appropriate path involving the least amount of data loss. Packets of data are delivered to target nodes based on a process of selection. Our proposed design is unique in a way as it involves no dedicated agents to choose paths. Each node comes equipped with self-servicing and self-aware

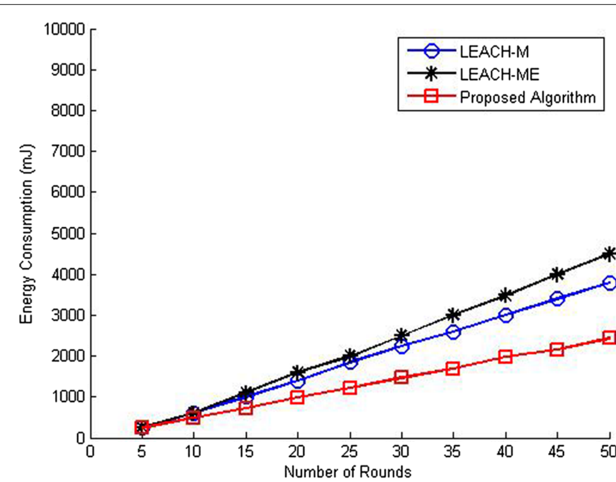

Fig. 9 Average energy consumption rate versus number of rounds

characteristics. As a result, the performance of our algorithm is expected to be more advantageous in comparison to cluster-based routing algorithm since the energy consumption due to cluster head selection is avoided. Without the overhead of the process of cluster-head selection, the efficiency of our proposed design is improved. For a smart-forest to evolve, our proposed wireless sensor, if rendered operable, will provide copious amounts of data with lower rates of data loss. Forest fires can be detected in initial stages and quickly triangulated by researchers and experts before these have the chance to cause widespread and devastating damages to the local ecosystem.

Abbreviations

EFR: experimental forests and ranges; ENQUIRY_ACK: enquiry acknowledgement; FWI: fire weather index; MICA: named after the mineral of the same name this open source hardware and software platform to provide sensing and computing in a single integrated package was designed by UC Berkeley and collaborating researchers; PDR: packet delivery ratio; RREP: route reply; RREQ: ROUTE REQUEST; SSEER: self-servicing energy efficient routing strategy; USDA: United States Department of Agriculture; WSN: wireless sensor network.

Authors' contributions

SR, RB and DS contributed to this work in the manuscript preparation. All authors read and approved the final manuscript.

Authors' information

Sandip Roy is currently pursuing Ph.D from University of Kalyani. He is an Assistant Professor in the Department of Information Technology, Brainware Group of Institutions, Kolkata, West Bengal, India. He has completed M.Tech in Computer Science & Engineering from HIT under WBUT in 2011. He has also done his B.Tech in Information Technology from WBUT in 2008. His main areas of research interest are Cloud Computing, Data Structure and Algorithm. Rajesh Bose is currently pursuing Ph.D. from Kalyani University. He is an IT professional employed as Senior Project Engineer with Simplex Infrastructures Limited, Data Center, Kolkata. He received his degree in M.Tech. in Mobile Communication and Networking from WBUT in 2007. He received his degree in B.E. in Computer Science and Engineering from BPUT in 2004. He has also several global certifications under his belt. These are CCNA, CCNP-BCRAN, and CCA (Citrix Certified Administrator for Citrix Access Gateway 9 Enterprise Edition), CCA(Citrix Certified Administrator for Citrix Xen App 5 for Windows Server 2008). His research interests include cloud computing, wireless communication and networking. Debabrata Sarddar Assistant Professor in the Department of Computer Science and Engineering, University of Kalyani, Kalyani, Nadia, West Bengal, INDIA. He has done Ph.D. at Jadavpur University. He completed his M. Tech. in Computer Science and Engineering from DAVV, Indore in 2006, and his B.E. in Computer Science and Engineering from NIT, Durgapur in 2001. He has published more than 75 research papers in different journals and conferences. His research interest includes wireless and mobile system and WSN, cloud computing.

Acknowledgements

Authors gratefully acknowledge to CSE Department of University of Kalyani, Brainware Group of Institutions, Kolkata, West Bengal, India, and Simplex Infrastructures Ltd, Data Center, Kolkata, for providing lab and related facilities for do the research.

Competing interests

The authors declare that they have no competing interests.

Received: 7 December 2015 Accepted: 7 April 2016
Published online: 19 April 2016

References

Allen R (2011) Seconds before the big one. In: Earthquake detection systems can sound the alarm in the moments before a big tremor strikes-time enough to save lives. ScientificAmerican.com

Allsopp M, Santillo D, Johnston P (2012) Climate Change impacts on arctic wildlife. Greenpeace Research Laboratories, Technical Report (Review), pp 1–20

Anitha RU, Kamalakkannan P (2013) EEDBC-M: enhancement of leach-mobile protocol with energy efficient density-based clustering for mobile sensor networks (MSNs). Int J Comput Appl 74(14):19–27

Basha E, Rus D (2007) Design of early warning flood detection systems for developing countries, In Proceedings of the conference on information and communication technology and development

Biet N (2014) Overview of a market that offers new opportunities to startups. THE FAKTORY, pp 1–25

Bouabdellah K, Noureddine H, Larbi S (2013) Using wireless sensor networks for reliable forest fires detection. Procedia Computer Science 19:794–801

Bradley J, Barbier J, Handler D (2013) Embracing the internet of everything to capture your share of $14.4 Trillion. In: Cisco and/or its affiliates, White Paper, pp 1–18

Casey K, Lim A, Dozier G (2008) A sensor network architecture for tsunami detection and response. Int J Distrib Sens Netw 4:27–41

Dyo V, Ellwood SA, Macdonald DW, Markham A, Mascolo C, Pasztor B, Scellato S, Trigoni N, Wohlers R, Yousef K (2010) Evolution and sustainability of a wildlife monitoring sensor network. In: Sensys'10, pp 127–140

Fraunhofer-Gesellschaft, "Research News-Fraunhofer" (2011)

Garcia-Sanchez AJ, Garcia-Sanchez F, Losilla F, Kulakowski P, Garcia-Haro J, Rodríguez A, López-Bao JV, Palomares F (2010) Wireless sensor network deployment for monitoring wildlife passages. Sensors 10:7236–7262

Hefeeda M, Bagheri M (2008) Forest Fire modeling and early detection using wireless sensor networks. Adhoc Sensor Wireless Networks 7:169–224

Heinzelman WB, Chandrakasan AP, Balakrishnan H (2002) An application-specific protocol architecture for wireless microsensor networks. IEEE Trans Wireless Commun 1(4):660–670

Hirsch K, Kafka V, Tymstra C, McAlpine R, Hawkes B, Stegehuis H, Quintilio S, Gauthier S, Peck K (2001) Fire-smart forest management: a pragmatic approach to substainable forest management in fire-dominated ecosystems. The Forestry Chronicle 77(2):357–363

Kim DS, Chung YJ (2006) Self-organization routing protocol supporting mobile nodes for wireless sensor network. In: Proceedings of the first international multi-symposiums on computer and computational sciences (IMSCCS'06), vol 02, IEEE Computer Society Washington, DC, USA ©2006 ISBN:0-7695-2581-4, pp 622–626

Kugker Z, De Groeve T (2007) The global flood detection system. In: JRC Scientific and Technical Reports, pp 1–45

Kumar GS, Vinu Paul MV, Athithan G, Jacob KP (2008) Routing protocol enhancement for handling node mobility in wireless sensor networks. In: TENCON 2008 IEEE Region 10 Conference, on pp 1–6

Lo S-W, Wu J-H, Lin F-P, Hsu C-H (2015) Cyber surveillance for flood disasters. Sensors 15:2369–2387

Lorincz K, Malan DJ, Fulford-Jones TRF, Nawoj A, Clavel A, Shnayder V, Mainland G, Moulton S, Welsh M (2004) Sensor networks for emergency response: challenges and opportunities. IEEE Pervasive Comput 3:16–23

Lyon LJ, Huff MH, Hooper RG, Telfer ES, Schreiner DS (2000) Wildland fire in ecosystems. In: General Technical Report, vol 1

Markham A (2008) On a wildlife tracking and telemetry system: a wireless network approach. In: Ph. D Thesis, Department of Electrical Engineering, University of Cape Town

Pasi AA, Bhave U (2015) Flood detection system using wireless sensor network. IJARCSSE 5(2):386–389

Rustad L, Campbell JL, Randall J, Adams MB, Schuler T, Hallett R, Hollinger D, Kabrick JM, Sebestyen S, Martin M, Brissette J (2014) Environmental sensor applications at USDA forest service experimental forests: the smart forest network. ASA, CSSA and SSSA international annual meeting, Long Beach, CA

Rustad L, Stine P (2013) Smart forests for the 21st century: a survey of cyber infrastructure capacity and needs at USDA forest service experimental forests and ranges. In: EFR "Smart Forests" Working Group

Sarddar D, Bose R, Sahana S (2015) A novel approach on weight based optimized routing for mobile cloud computing. Braz J Sci Technol 2(3):1–12

Shnayder V, Hempstead M, Chen B, Allen GW, Welsh M (2004) Simulating the power consumption of large-scale sensor network applications. In: Proceedings of the second ACM conference on embedded networked sensor systems (SenSys'04), Baltimore, MD

Tan R, Xing G, Chen J (2013) Fusion-based volcanic earthquake detection and timing in wireless sensor networks. ACM Trans Sensor Netw 9(2):17

The Consititution of India, 1st December 2007, p 24

Trifa VM, Girod L, Daniel TC Blumstein T, Taylor CE (2007) Automated wildlife monitoring using self-configuring sensor networks deployed in natural habitats. In: Proceedings of the twelfth international symposium on artificial life and robotics

Yavasli DD, Tucker CJ, Melocik KA (2015) Change in the glacier extent in Turkey during the Landsat Era. Remote Sens Environ 163:32–41

Zhu P, Jia F (2012) A novel hybrid self-organizing clustering routing algorithm. Int J Distrib Sensor Netw 2012:146516

Zhu Y, Xie L, Yuan T (2012) Forest fire detection system based on ZigBee wireless sensor network. In: ICEICE'12 proceedings of the 2012 second international conference on electrical information and control engineering, vol 2, pp 761–764