Advances in Computer Science and Information Technology (ACSIT)
Print ISSN : 2393-9907; Online ISSN : 2393-9915; Volume 2, Number 7; April – June, 2015 pp 64 – 67
Krishi Sanskriti Publications
http://www.krishisanskriti.org/ACSIT.html

Bee Inspired Routing Protocol for WSN with Cross Layer Architecture

Neeraj Kumar[1], Rajeev Mohan Sharma[2]

[1]Department of Computer Engineering
National Institute of Technology
Kurukshetra, India
[1]neerajkadyan205@gmail.com
[2]Department of Computer Engineering
National Institute of Technology
Kurukshetra, India
[2]rmsharma123@rediffmail.com

Abstract: *A wireless sensor network consisting of a number of sensor nodes provides to be an efficient tool for collecting data in a variety of situations like forest under fire, war like situations etc. the main concern is to develop an effective energy efficient routing protocol so as to have a good impact on overall life time of sensor network. However finding an optimal path in wireless sensor network (WSN) is a difficult issue because of the limited resources and their dynamic nature. Nowadays, most of the research is focused on bio-inspired systems; the reason behind this is their ability of adaptation, scalability, self-organization, robustness and decentralized control. This paper examines many bio inspired routing algorithms, along with a new proposed algorithm which is based on the foraging principle of honey bees. Our proposed algorithm is a reactive and multipath routing protocol for WSNs.*

Keywords: *reactive protocol; bee colony algorithm; cross layer design; routing protocol.*

1. INTRODUCTION

A wireless sensor network contains a number of sensor nodes and sink nodes. WSN has the ability to sense the physical conditions like pressure, humidity etc., and communicate with neighboring nodes for further processing over collected data. WSN has shown a remarkable impact on military and civil application like detecting position of terrorist in the dense forest area, disaster management, war field surveillance, security etc. A sensor node basically constitutes of four parts namely: Sensing unit, Processing unit, Transceiver and Power unit.

In WSN, routing process is a complicated due to limited resources like low computational power, low power etc. The idea behind every routing technique is to enhance the network life time.

WSNs faces various challenges like: limited energy capacity, nodes deployment, location of sensors, latency, changing topology, storage limitation, limited processing, low computational power, scalability and fault tolerance. The routing in WSN faces various challenges. These are: (1) forming a global addressing scheme with a large number of sensors is quite difficult task, (2) facing redundancy that occurs due to same data generated by on the network.

If we try to percept through our nature it provides various solution and inspiration to solve certain problems like wind turbine design, bullet train, aircraft wing design and routing techniques also. Many protocols have been devised through the foraging behavior of insects like termites, ants, honey bees, etc. The nature inspired algorithms has been of immense importance due to their ability of discovering better solutions for routing problems and has attracted the attention of researchers and engineers. The basic bio-inspired approaches have the ability to find the shortest path, however sometimes it might not give the minimum energy cost path. So as a solution to this problem we can make use of cross layer approach.

In TCP/IP network model, the communication was allowed only between alongside layers which proves to be a limitation and results in some side effects like compromise of quality if service (QoS), delay, extra overhead etc. As a solution to this problem cross layer design has proved to be a better alternative and we can communicate directly with others layers as well. However, Cross layer design doesn't demolish the structure of TCP/IP network model.

The paper is structured as follows: Section II describes the related research, section III shows our proposed scheme and the last section concludes the paper.

2. RELATED RESEARCH

In this section, we provide a literature review of existing routing algorithms. Table 1 shows the comparison between the existing routing protocols.

A. Classical routing algorithms

In general, routing protocols in WSNs comes under proactive and reactive techniques. In proactive, nodes maintain at least one routing table and whenever a path is required, nodes run a shortest path finding algorithm to find a path. Destination sequenced distance vector routing protocol (DSDV) comes under proactive routing protocol whose idea has been taken from Bellman Ford algorithm. In reactive protocols, routes are discovered when required through a route discovery process. Reactive protocols may also be termed as on-demand protocols.

Energy efficient Cross-Layer routing algorithm (E2CL) [1] has been identified as a reactive protocol that rely on inter-communication between non-adjacent layers in an attempt to increase the lifetime of wireless sensor network. E2CL basically subsists of three states namely route discovery, route maintenance, and route re-establishment. Through route discovery one can find an optimal path by using route-request packets and reply packets. The decision is taken by using a cost function, which further is based upon following three parameters: power conservation, potential interference between nodes and distance among nodes. While doing route maintenance, if either link cost goes beyond the threshold value or when it is unable to find next-hop neighbor node then in both cases node sends a request to its preceding node, then preceding node replies with a route recovery message. Whenever the lost link is not recovered by route maintenance, route re-discovery request message is sent back to the source node which initiates the route discovery process. The advantage of using E2CL over DSR (Dynamic Source Routing) is that it increases the lifetime of the network.

B. Bio-inspired routing protocols

Bio-inspired routing protocols explain the routing mechanisms that are based on the social behavior of insects. M. Dorigo et al. [3] proposed first ACO based routing protocols. ACO algorithms are inspired by real ant behavior for solving various complex problems. For information gathering and path finding it makes use of forward ants (FANT) and backward ants (BANT). The information exchange is done by using pheromone value like real ants.

Ant Routing Algorithm (ARA) [2] is categorized as on-demand and multipath routing protocol, which makes use of ACO algorithm. Here also route selection is accomplished by forward and backward ant agents. Route maintenance is also done by using data packets. If a node detects a route failure on getting a missing acknowledgement, then it searches for some path in its routing table otherwise it makes use of backtracking (sending a route error message to its preceding node and so on). Source node on receiving route error packet will starts the route discovery process. The benefits of using ARA over AODV and DSDV (Destination Sequenced Distance Vector) are: enhanced throughput decreased routing overhead and improved network lifetime.

The disadvantage with Sensor-driven Cost aware Ant Routing (SC), Adaptive ant-based Dynamic Routing (ADR), E&D ANTS, AR (Adaptive Routing) and IAR (Improved Adaptive Routing) was that these protocols were unable to determine minimum energy cost route. In 2009 K. Saleem et al. [3,4] discovered a cross layer based routing protocol for WSN to solve the above problem. In his technique the route exploration was carried out by using ACO and the routing decision was made by using parameters like energy level, link quality and velocity. Where above parameters were determined by the intercommunication between physical layer and the network layer. This algorithm proved itself as a better solution to deadlock problem by giving a sequence ID to each ANT. This algorithm was able to find the minimum energy route quite efficiently and gives better throughput.

Dervis Karaboga [5] in 2005 proposed a new routing protocol which is based on the social behavior of bees known as BCO. The employed and onlooker bees find new sources based on their experience and scouts move randomly for finding new food sources. The information exchange between bees is done by waggle dance.

BeeAdHoc [6] is energy efficient, reactive and source routing algorithm (Each packet header contains the whole path to the destination having node IDs of each and every node from source to destination) in mobile ad-hoc networks (MANETs). The idea of BeeAdHoc has been derived from the foraging principle of honey bees. The algorithm includes four kinds of bee agents: packers, scout, foragers and swarms. Packers are responsible for finding a forager for data packets and they die instantly after handover it to foragers. Scouts explore the path by making use of broadcasting principle. After finding path scouts transfer the whole information to foragers regarding path to be followed and then foragers transmit the data packets to the destination. As soon as all the forgers reach the destination, they all are sent back to its source node by using swarms. All such foragers on reaching destination are referred as waiting foragers. In BeeAdhoc, while sending back waiting foragers, one forager is placed in header while remaining are placed in the payload part of the swarm and send to the source node. A node in BeeAdHoc contain three sub-parts: packing floor, entrance and dance floor. The packing floor provides an interface to the transport layer. Entrance provides an interface to the MAC layer. The routing decision happens on the dance floor. In BeeAdHoc, the validity of the path is being controlled by the number of foragers. When compared to DSR, AODV and DSDV, it consumes less energy.

BeeIP [7] is an on-demand and multipath routing in MANETs. In BeeIP, on route requirement the source node generates a scout packet and broadcast the scout in the whole network as it does not hold any previous information. The scout packet determines the available paths to the destination. The desired destination on receiving scout generates an ack_scout packet and unicast this to the source. The transmission of data packets is being carried out by foragers. Ack_scouts continuously over a period of time gather the low-level parameters like velocity, signal strength, transmission delay, size of the MAC queue and remaining energy by using cross layer architecture. BeeIP technique calculates the path quality with the help of above mentioned factors, whereas path quality in BeeAdHoc, consider the parameters such as: remaining energy and delay between the links. The effectiveness of this protocol lies in end to end delay and packet delivery ratio.

Table 1 COMPARISON BETWEEN ABOVE ROUTING PROTOCOLS

Routing algorithms	Categorization	Parameters	Network type	Reported strength
E2CL	Classical	Distance, signal strength, energy	WSN	Energy efficient
ARA	ACO	Pheromone value	MANET	Less overhead
K. Saleem	ACO	Energy, delay, velocity	WSN	Data loss ,throughput
BeeAdHoc	ABC	Energy, delay	MANET	Energy efficient
BeeIP	ABC	Velocity, signal strength, energy, size of MAC queue	MANET	PDR, delay

3. PROPOSED SCHEME

Our proposed algorithm is a on-demand and multipath routing algorithm for WSNs. This algorithm is based on the social behavior of honey bees. There are two types of bee agents: scouts and foragers. Scouts discover multiple paths between sources and sink node and also calculate the quality of path. A scout makes use of broadcasting principle with a time to live timer (TTL) field, so as to limit the life span of scout to avoid its indefinite circulation. Foragers are transfers the data packets from source to sink node. The proposed scheme subsists of two phases: route discovery and route maintenance. In the first phase we discover multiple paths between sensors and sink node. In the second phase we maintain the paths which are broken due to the dynamic nature of WSNs.

C. Route Discovery

Assume that the entire network has been initialized and each node has its own ID. For route discovery, we use two types of scouts: for_scout (forward scout) and back_scout (backward scout). The for_scout traverse the network to find a sink node. Once a sink node found, the back_scout establish multiple routes between sources and sink node.

Initially, when a path is required between source sensor and sink node and source don't contain any previous information, a for_scout (Fig. 1) packet is created and broadcast to the network. When a node *j* receives a for_scout from a node *i* for the first time, then intermediate node *j* decide whether the for_scout packet is broadcast or not. If intermediate node have information (i.e. path), it unicast the for_scout and update the path list and quality field value. Otherwise it does the following things:-

- Add its node ID to the path list.
- Gather the low level information i.e. remaining energy, speed and signal strength.
- Calculate the link quality between nodes by using the equation 1 and update the quality field of scout.

$$q_{ij} = w_e * re + w_{speed} * speed + w_{sig} * sig \quad (1)$$

Where q_{ij} stands for link quality between node i and node j, *re* stands for remaining energy, sig stands for signal strength, w_e, w_{speed} and w_{sig} are appropriate weight constants.

***In order to be used effectively, the above low level parameters are to be normalized, which is done by applying linear transformation.

- Wait for a period of time.

During this period, intermediate node collects the for_scouts which belong to the same source and calculate the quality parameter. The for_scout which have greater value of quality q_{ij} is to be further forward and rest of the for_scouts are discarded. Each intermediate node maintains a link quality table, which contains the quality of the link between its neighbors. Link Quality table contain two fields i.e. neighbors node ID and quality value. Each intermediate node repeats the same step till the for_scouts reached to the sink node.

Source ID	Scout ID	TTL	Path List	Quality

Fig. 1 The For_scout packet

Source ID	Scout ID	PID	Path List	Quality

Fig. 2 The Back_scout packet

Node ID	Quality

Fig. 3 Link Quality Table

Source ID	Sink Node ID	PID	Path List	Quality

Fig. 4 Routing Table

When the first for_scout reaches to sink node, it generate the back_scout (Fig. 2) packet and starts a timer. During this period of time collects all the for_scouts packets and according to the quality of their path (Q_P), sink node give the identification number (rid) to each discovered route. The path list fields of the for_scouts are copied reverse in the path list field of back_scouts. Instead of broadcasting, the back_scouts are unicast. The quality of the entire path can be expressed as:

$$Q_P = \sum_{n=1}^{m-1} (q_{N_{n+1} \to N_n}) \ [source = N_1, sink\ node = N_m] \quad (2)$$

Where Q_P is the path quality between a source and sink node, m is the total number of nodes in the entire path and $N_{n+1} \rightarrow N_n$ is the pair of nodes.

When the back_scout packet reaches the source, it creates a routing table. Once a route is discovered, data is transfer from source to the sink node with the help of foragers. The proposed algorithm discovers multiple paths so data packets are distributing on multiple paths for load balancing.

In this paper, we gathered low level information i.e. remaining energy, speed of node and signal strength by using cross layer design. In layered architecture, the layers are isolated from each other. Due to this some side effects comes, including compromise of QoS, extra overhead etc. Therefore, to overcome the side effects of layered architecture, cross layer design have been proposed. Anything that violets the layer structure is known as cross layer design. Cross layer design don't destroy the layer architecture, but provide the communication between the layers. In the physical (PHY) layer the received signal quality is determined at receiver by using signal to interference ratio (SIR), where SIR is calculated as the power of wanted signals to the total residue power of unwanted signals. The information of speed and remaining energy is given by application layer.

D. Route Maintenance

In WSN, nodes are mostly mobile and die due to limited power source. Because of above reason the topology of the WSN changes quite randomly. It is not always good to re-establish the lost link every time by using route discovery phase. Each intermediate node constantly monitors its quality table. When the quality of a link goes below a threshold value, the intermediate nodes send a path error (PERR) message to the source. Similarly, when a node can't find its next-hop neighbor, it sends a PERR message to the source. Thus the proposed protocol recovers the broken link in a effective way.

4. CONCLUSION

In this paper we presented a new approach for routing for wireless sensor networks. Our proposed algorithm is a on-demand, source routing (Each packet header contains the whole path to the destination having node IDs of each and every node from source to destination) and multipath routing algorithm for WSNs. Our approach is based on honey bees and aim to enhance the network life time and reduce the routing overhead. In proposed scheme, cross layer architecture is used for collecting low level parameters.

Our proposed algorithm can be implemented in MatLab or Network Simulator (ns-2). Our future plan is to implement the algorithm in the ns-2 and compare the results with other routing protocols. For simulation these parameters can be used: delay, throughput, data processing cycles per node, overhead, etc.

REFERENCES

[1] Lu Xianling, S. Yaming, Yu Jiming, Y. Wenxu, "An energy efficient cross-layer routing algorithm for wireless sensor networks," Industrial Electronics and Applications, 2006 1st IEEE Conference in Singapore, pp. 1-5.

[2] M. Gunes, U. Sorges, I. Bouazizi, "ARA- The ant colony based routing algorithm for MANETs," Proceedings of the 2002 ICPP Workshop on Ad Hoc Networks (IWAHN 2002), pp. 79-85, IEEE, Augest 2002.

[3] K. Saleem, N. Fisal, S. Hafizah, R. Rashid, Y. Baguda, "Cross layer based biological inspired self organized routing protocol for wireless sensor network," TENCON 2009 – 2009 IEEE Region 10 Conference, pp. 1-6.

[4] K. Saleem, N. Fisal, S. Hafizah, R. Rashid, Y. Baguda, "A self optimized multipath routing protocol for wireless sensor networks," International Journel of Recent Trends in Engineering, vol. 2, No. 1, November 2009.

[5] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Technical report-tr06, Erciyes university, engineering faculty, computer engineering department, vol. 200, 2005.

[6] H. F. Wedde, M. Farooq, T. Pannenbaecker, B. Vogel, C. Mueller, R. Jerushkat, "BeeAdHoc: An energy efficient routing algorithm for mobile ad hoc networks inspired by bee behavior," in: GECCO'05 Proceedings ot the 7th annual conference on Genetic and evolutionary computation, pp. 153-160, 2005.

[7] A. Giagkos, M. S. Wilson, "BeeIP: A swarm intelligence based routing for wireless ad hoc networks," Information Sciences, vol. 265, pp. 23-35, May 2014.