IJASRM
International Journal of Advanced Scientific Research and Management, Volume 4 Issue 1, Jan 2019
www.ijasrm.com
ISSN 2455-6378

Wireless Sensor Networks Based on Shortest Path Algorithms

Aijaz Magray[1], Mudasir Younis[2,*], Chitaranjan sharma[2]

[1] Department of Mathematics, Holkar Sciences College, Devi Ahiliaya University, Indore (M.P), 452001, India

[2] Deapartment of Applied Mathematics, UIT-Rajiv Gandhi Te chnological University,
Bhopal, M.P., 462033, India

[3]Department of Mathematics, Holkar Sciences College, Devi Ahiliaya University, Indore (M.P), 452001 India

Abstract

Since wireless networks grants so many encounters to standard routing algorithms. Wireless networks communication comprises huge number of sensor nodes i.e. which transmit and receiving protocol. Since sensor nodes has a specific range to send the protocol to its destination and hence the protocol is sent by the use of routing algorithm (shortest path algorithm) but suitable algorithm should be must accurate in operation, stability, simplicity, equality and optimality. Therefore the objective of this paper is to perform shortest path algorithm in wire communication network which will help to transmit protocol and should reach its destination in real time so to cover and identity the shortest path algorithm in wireless communication system plays a major role for searching shortest distance in shortest possible time

Keywords: *Shortest path, protocol, sensor nodes, Dijkstra's Algorithm, Bellman ford*

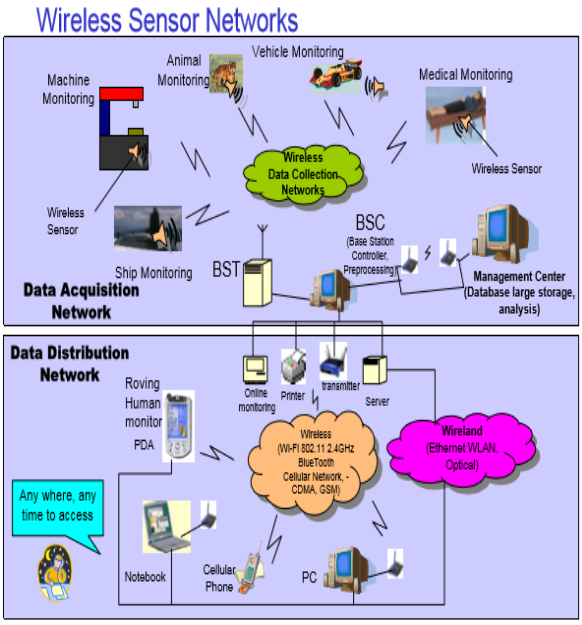
1. Introduction

Wireless sensor networks (WSNs) can be defined as the self-configured and infrastructure less wireless network to monitor physical and environmental conditions, such as temperature, motion pressure, vibration, sound, or pollutants and to cooperatively pass their data through the network to the main location or sink where the data can be observed and analyzed. A sink or base station acts like an interface between the network and the user. One can retrieve required information from the network by injecting queries and gathering results from the sink. Typically a wireless sensor network contains hundreds, thousands of sensor nodes [1]. The sensor nodes can communicate among themselves using radio signals. A wireless sensor node is equipped with the sensing and computing devices, radio transceivers and power components. The individual nodes in a wireless sensor network (WSN)[2] are inherently resource constrained; they have limited processing speed, capacity, storage, and communication bandwidth. After the sensor nodes are deployed, they are mostly

Responsible self-organizing an appropriate network infrastructure often with multi-hop communication with them. Then the on board sensor starts collecting information of interest

Wireless sensor devices also respond to queries sent from a 'control site' to perform specific instructions or provide sensing samples. The working mode of the sensor nodes may be either continuous or event driven. global positioning system [GPS] and local positioning algorithms can be used to obtain location and positioning information .Wireless sensor devices can be equipped with actuators to 'act' upon certain conditions. These networks are sometimes more specifically referred as wireless sensor and actuator network.

IJASRM
International Journal of Advanced Scientific Research and Management, Volume 4 Issue 1, Jan 2019
www.ijasrm.com
ISSN 2455-6378



Wireless sensor networks [WSNS] enable new applications and require non-conventional paradigms for protocol design due to several constraints. Owing to the requirements for low device complexity together with low energy consumption (I.e. long network life time), a proper balance between communication and signal/data processing capabilities must be found this motivates a huge effort in research activities, standardization process ,and industrial investments on this field since the last decade at prevent time, most of the research is an [WSNs] has concentrated on the design of energy and computationally efficient algorithms and protocols and the application domain has been restricted to simple data-oriented monitoring and reporting applications [3,4,5]

Currently Wireless communication system is developed continuously but several factors significantly affect the wireless market. One of them is that customer expectations have risen to demand high data rate multimedia applications for example video applications. The second one factor is the rapid development of multimedia technology and internet. 3G is a new standard that rise to challenge these factor services in the 3G were broadened from regular telephone services to include high speed internet service and video communication. Now industry migrates 3G into 4G wireless system designers facing two significant technical challenges

- Bandwidth limitation
- Effect of Multipath fading

In this research we have perform routing in Wireless communication network by the use of Dijkstra's and Bellman ford algorithms which will solve the major problem of fading in Wireless communication system. In Wireless communication when we transmit a protocol we need to reach it another node in shortest possible time in order to decrease the fading. Because fading will be major if the whole work of transmitted protocol should not reach its destination in real time. So to cover and identify the shortest path in Wireless communication system plays a major role which can be improved by using the Dijkstra's and Bellman ford algorithm

2. Classical approach

While it will be shown that the following static routing algorithms are not appropriate for wireless networks in general, the underlying methods and objectives provide the foundation to which several effective routing protocols are based on [6]

(a)Link-state protocol; Dijkstra's Algorithm;

The algorithm presented below produces the shortest path from the source s to all possible destinations v on the directed graph G in an non-negative edge weights w. its implementations in networking is known as a link-state algorithm [7,8] . let |G| denotes the number of nodes in the graph. let D(v) be the total weight of the shortest path from the source s to destination v. let C (u, v) be the cost of routing from u to v I.e. the sum of all edge weights along the path from u to finally let N contain the vertices in G whose shortest path have been determined

Algorithm Dijkstra's Ls (G,W,S)

A graph G (represented by an adjacency list), non-negative edges weights W, and source s

Output:

Weight of shortest path from source s to all destination v

1 N = [S]

2 for all destinations $V \in G$

3 if V is adjacent to S

4 then $D(V) = C(S, V)$

5 else $D(V) = \infty$

6 for $i = 1 \; to \; |G| - 1$

7 Amongst all nodes $n \notin N$ adjacent to any

8 $V \in N$, add n to N such that

9$D(V) + C(V, n)$ is a minimum

The structure of the above algorithm comes from [9] while a more detailed analysis of its functionality and correctness is outlined in[10] lines 6 through 9 are known as relaxing an edge. A total of 1g1-1 edges are relaxed, each requiring o(1G1) computations, which results in an all,over complexity of $o|G^2|$. the implementation of a binary heap For the priority queue improve the relaxation computing to o[log 1G1] and thus the overall complexity to o[1G1 log 1g1]

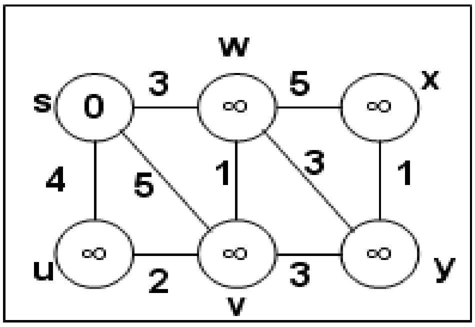
Regardless, the implementation of link–state protocols generally requires global state information from the graph. That is the topology and link-costs are known to every node, which culminates in one exhaustive routing table that is maintained between all nodes. Hence small change in the topology of the

IJASRM
International Journal of Advanced Scientific Research and Management, Volume 4 Issue 1, Jan 2019
www.ijasrm.com
ISSN 2455-6378

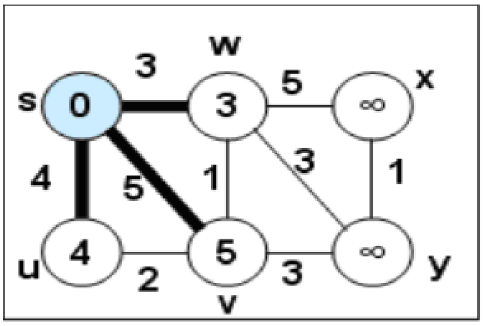
system can be propagate into large errors in the routing table. Even with the available global state information, routing loops can still occur, for instance when two nodes a and b attempt to route through each other to get to a destination d, an example of this event will be provided in the next section, as the direct distance vector routing protocol is also sensitive to change in topology. Thus the use of Dijkstra's algorithm in wireless ad hoe network is ill suited due to its static nature , time-complexity and the overhead requirements it imposes on a network by requiring global state information

3. Directed Distance Protocol: The Bellman-Ford Algorithm

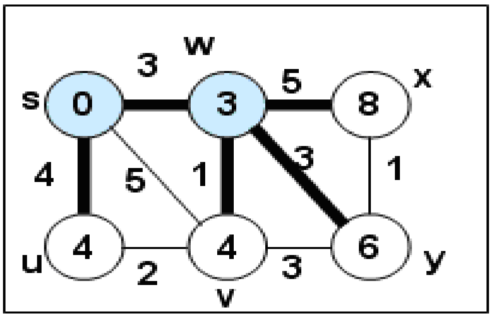
The direct distance –vector routing protocol is necessary identical to the bellman–ford algorithm but with one modification ; the relaxation phase uses an infinite while loop in order to achieve a quiescent state that responds to link-cost changes or updated distance vectors [11] A major distinction between link state and directed distance –vector routing algorithm is the amount of information that is available to a given in a network .the algorithm below uses localized information of a source to calculate the distance vector to each of its neighbors once each node has calculated its distance vector , it is broadcasted to each of their respective neighbors. the results is minimum weight –spanning tree The run time complexity of the Dijkstra's shortest path is $o(|v|^2)$



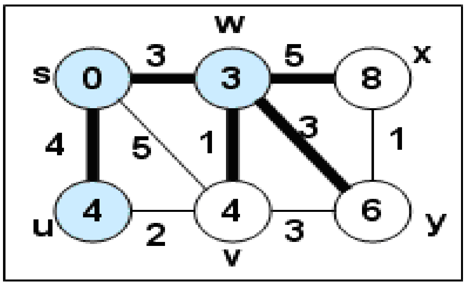
Initialization



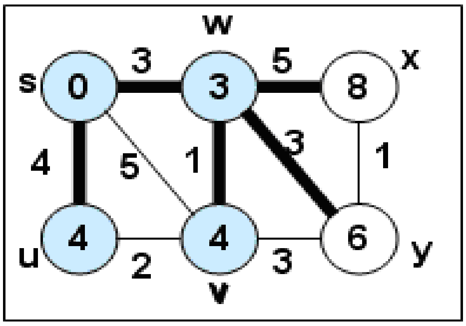
Iteration 1



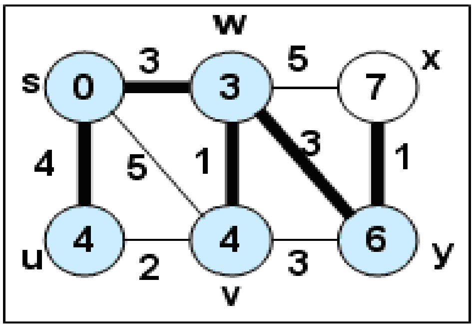
Iteration 2



Iteration 3



Iteration 4



Iteration 5

Graph in wireless communication system

4 Mathematical description of the algorithm

Step 1 put Vs into M, Step 1: choose Vj from M so that Dist[j] = min{Dist[m] / m ∈ M}**…. 1**

Vj equals to end node, then end.

Step 2 find Vk that is linked with Vj in R,then

Dist[Vk] = min{Dist[Vk],Vk+Cost(Vi;Vj)} **…….. .2**

put it into M, then go Step 1

.

5 Algorithm Bellman –Ford DDV [G,W.S]

We now turn to solving the single source shortest path problem in the general case where we allow negative weights in the graph. One might ask how negative weights make sense. If talking about distances on a map, they probably do not, but various other problems reduce to shortest paths, and in these reductions negative weights show up. Before proceeding we note that if there is a negative weight cycle (the sum of weights on the cycle is negative) reachable from the source, then there cannot be a solution. This is because every time we go around the cycle we get a shorter path, so to find a shortest path we would just go around forever. In the case that a negative weight cycle can be reached from the source vertex, we would like solutions to the SSSP problem to return some indicator that such a cycle exists and terminate

Input:

A graph G (represented by an adjacency list) edge weights W, and since s,

Output;

Weights of shortest path from source s to algorithm v

1 $D(S) = 0$
2 for all the destinations $V \in G$
3 if V is adjacent to S
4 then $D(V) = C(S,V)$
5 send $D(V)$ to all adjacent nodes V
6 else $D(V) = \infty$
7 while (there exists a link cost change or update distance vector to same adjacent V)
8 do; for all $n \in G$
9 $D(n) = \min\{D(n), D(v) + C(V,n)\}$
10 send update distance vector; $D(V) = [D(V); adjacent\ V \in G]$ to all adjacent nodes V
11 if $D(S) > D(V) + C(V,S)$
12 then report negative weight cycle
13 Break

Figure 1 illustrates how the bellman–ford algorithm produces the shortest path in the given graph with A as the source. FIGURE 2 exemplifies the distributes and asynchronous nature of the algorithm where each node maintains its own routing table and sends updated distance vector to neighboring nodes

Note that the purpose of lines 14 to 16 is to cheek for negative weight cycles with in the graph. This may be unnecessary of all edge weights are non-negative, but depending on what the routing lost are representing negative weights may be unavoidable. for instance they might signify the load carried or removed from a packet as it travels from a source to a destination.it is simply one of many preventative measures against routing loops[14,]. Unfortunately the bellman –ford algorithm can converge very slowly, and despite its ability to accept changes in topology it is prone to routing loop[12,13,] An example of how a link –cost change can result in a routing loop is showing in fig.3

In the first picture of Figure 3, the shortest path has already been determined between nodes, AA, BB and CC. The problem arises when the link cost between nodes BB and CC changes from 5 to 20. At this moment, BB detects the link-cost change and updates its distance vector accordingly to route through AA to get to CC. But AA is not aware of the link-cost change and will continue to route through BB to get to CC which results in a routing loop between AA and BB. However, through continually updating their distance vectors, AA will eventually route to CC directly, terminating the routing loop. This phenomenon is known as the count-to-infinity problem [2].

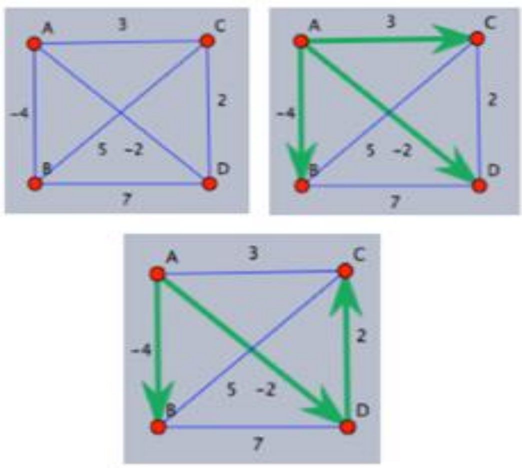


Fig 2 graph to find shortest path

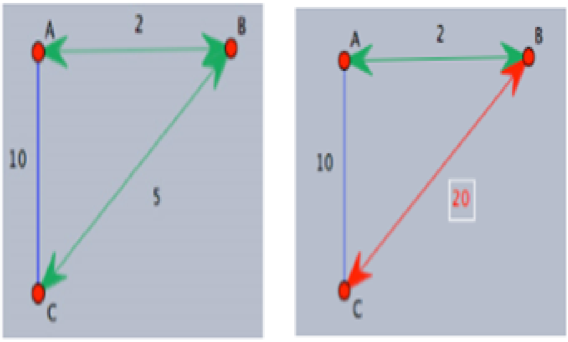


Fig 3 result in routing loop

Adding poisoned reverse is a technique designed to counteract this issue and can be applied to the example in Figure 3 in the following way: since the link-cost change results in BB attempting to route through AA to get to CC, BB will broadcast its cost of routing to CC as $DD(CC)=\infty$. When AA receives the updated distance vector from BB, it will now believe that a direct path from BB to CC does not exist and will therefore route to CC directly (as it would have all along if global state information was available. While localized information certainly decreases the overhead costs of routing protocols, it enables errors to propagate from node to node. Despite the existence of preventative measures such as adding poisoned reverse, Dijkstra's algorithm and the Bellman-Ford algorithm are neither robust nor computationally efficient enough to handle the dynamic nature of wireless ad hoc networks. In order to formulate routing protocols that are appropriate for these purposes, desirable structures will be analyzed next.

6. Conclusions

At first glance it word appear that the link state and distance-vector routing protocols are ill suited for wireless ad-hoc networks. But the majority of heuristic protocols used are largely influenced by their methodology. Which is highly undesirable of these methods are the overhead costs associated with maintaining routing tables

The unit distance wireless graph is a good representation of wireless ad hoc networks in which the wireless devices all have a uniform broadcast range of course there are many cases in which broadcast ranges are not uniform and there are often environmental factor that can inhibit a transmitter signal strength. However this model provides the foundation to which compass routing and more importantly face routing can guarantee delivery.

It was suggested that since nodes with in a unit distance wireless graph generally have relatively few adjacent nodes, the greedy quadratic time algorithms are better suited for determining incident edges, as there implementation is simpler. However, the robustness and scalability of face routing is undeniable and should at least be employed as an alternative routing protocol when routing loops persist with in a system

Acknowledgments

We would like to thank the anonymous referee for the valuable comments that has helped to improve the presentation of this article.

Reference

[1] Misra, P.(1999) Routing protocol for ad hoc mobile,wirelessnetworks.[online].Available,at:http|| www.cse.wustl.edu\jain|cis,788,99|ftp|adhoc_routing| index.html

[2] Kurose, j.Ross, K.(2012).computer networking: A top down approach.(6th ed.)Addison-wesley, Boston(MA).

[3].cormen,T.,Leiserson,c.,Rivest,R.(1999).introduction to algorithms page 126.The MIT press, Cambridge, MA.

[4] Urrutia,j.(2007).Local solutions for global problems in Wireless networks. Journal of Discrete Algorithms 5(3),395-407

[5] Y.C. Wang, W.C. Peng, and Y.C. Tseng, "Energy balanced Dispatch of Mobile Sensors in a Hybrid Wireless Sensor Network," IEEE Trans. On parallel and distributed systems, vol. 21, no. 12, December. 2010

[6] A. Verma, H. Sawant, and J. Tan, "Selection and Navigation of Mobile Sensor Nodes Using a Sensor Network,"Pervasive and Mobile Computing, vol. 2, no. 1, pp. 65-84, 2006.

[7] Y.C. Wang and Y.C. Tseng, "Distributed Deployment Schemes for Mobile Wireless Sensor Networks to Ensure Multilevel Coverage," IEEE Trans. Parallel and Distributed Systems, vol. 19, no. 9, pp. 1280-1294, Sept. 2008.

[8] Y.C. Wang, C.C. Hu, and Y.C. Tseng, "Efficient Placement and Dispatch of Sensors in a Wireless Sensor Network," IEEE Trans. Mobile Computing, vol. 7, no. 2, pp. 262-274, Feb. 2008

[9] L. P. Chew, "Planning the shortest path for a disc in O (n2 log N) time," in Proc. ACM Symp. Computational Geometry, 1985, pp. 214-219.

[10] Y. H. Liu and S. Arimoto, "Proposal of tangent graph and extended tangent graph for path planning of mobile robots," in Proc. IEEE Inr. Con$ Robot. Automat. 1991, pp. 312-317

[11] "Path planning using tangent graph for mobile robots among polygonal and curved obstacles," Int. J. Robot. Res., MIT Press, vol. 11, no. 5, pp. 376-382, 1992

[12] J. Hershberger and L. Guibas, "An O(n2) shortest path algorithm for a non-rotating convex body,'' J. Algorithms, vol. 9, pp. 1846, 1988

[13] Gene Eu Jan, Ming Che Lee, S. G. Hsieh, Yung-Yuan Chen,"Transportation Network Navigation with Turn Penalties", IEEE/ASME International