Real time Energy Efficient automated Industrial Equipment using Routing Technique in Wireless Sensor Network

Sudipta Priyadarshinee.

Assistant Professor in Indic institute of design and research, department of Computer Science .

ABSTRACT

Wireless sensor systems can give minimal effort answer for verity of real life issues. Sensors are minor gadgets with restricted capacity, computational ability and power. They can be conveyed in huge scope for performing both military and regular citizen tasks. Security will be one of the fundamental concerned when they will be sent in enormous scope. As sensors have constrained force and computational capacity, any security instrument for sensor arrange must be vitality effective and ought not be computational intensive.

A WSN comprising of dispersed self–composed independent gadgets worked with sensors to screen physical ecological conditions. We can also implement the Wireless sensor networks (WSN) to the industries for the betterment purpose. They are well known for mechanical detecting applications because of their generally ease and straightforwardness for retrofitting into existing framework. Wireless Sensor Networks has made a significant impact in the automation and control of the industrial processes. But while mounting the sensor in to the industries we also have to put our focus to the routing mechanism. Because: until and unless the routing technique is not correct we cannot achieve our goal. Similarly, the another important factor in WSN is Energy efficiency which is the most important issue in research of wireless sensor networks while the routing protocol plays an important role in achieving energy efficiency in WSN. Here we had taken the example of textile machinery where I supposed to put a sensor node at different stages of the machine. The main intension to do so is to automate the machine and to know the status of different phases at particular instance. As the status of each component in the machine shows the efficiency and hence the production cost of the company; i.e. if the exact status of the machine is known to us than we can compute the cost per production. Hence, we had taken this textile as my paper subject. Here I had used a Dijkstra's and prim's shortest path algorithm. Because: To make the production better with few expenses we need to have lesser number of components with multifunctional approach. If the total path travelled by the system is small than obviously the cost per production should be small. I had tried here to show the overall strategy in terms of graphs (mathematically). Hence using this algorithms I had tried to find the most optimal shortest path so that with less number of component and sensor node can reach to our target.

INTRODUCTION

Wireless Sensor Networks (WSN) have increased overall consideration in recent Years because of the advances made in remote correspondence, data innovations and hardware field .The idea of remote sensor systems depends on a straightforward condition: Sensing + CPU + Radio = Thousands of potential applications . It is a detecting innovation where little, minimized and self-governing gadgets called sensor nodes conveyed in a remote zone to distinguish phenomena, gather and procedure information and transmit detected data to clients. The advancement of minimal effort, low-power, a multi functional sensor has gotten expanding consideration from different ventures.

Application of Wireless Sensor Network.

The applications of wireless sensor network mainly include health, military, environmental, home, industry & other commercial areas.

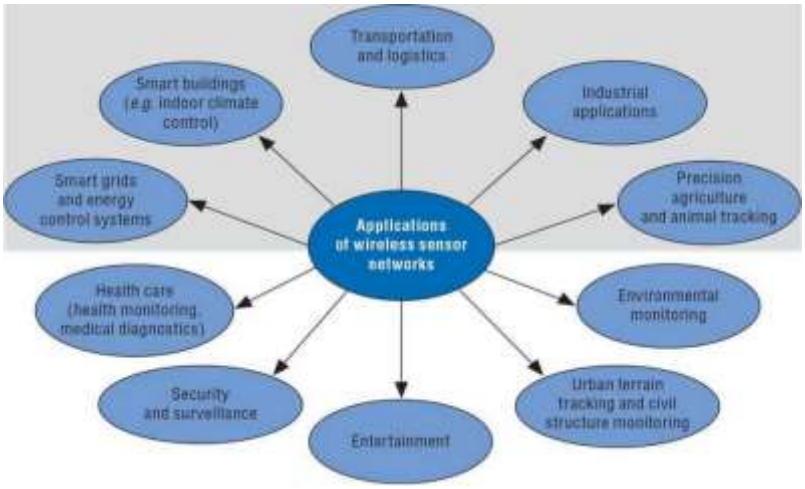
i. Military Application
ii. Home Applications
iii. Industrial Monitoring
iv. Health care monitoring
v. Environmental Applications

vi. Commercial Applications

vii. Air pollution monitoring

viii. Forest fire detection

ix. Landslide detection



Characteristics of Wireless Sensor Network

The characteristics of WSN include the following:-

I. Dyanamic network topology

II. Self organistaion.

III. Low cost.

IV. Capacity to handle with node failure

V. mobility of nodes

VI. Heterogeneity of nodes

VII. Scalability to large scale of distribution

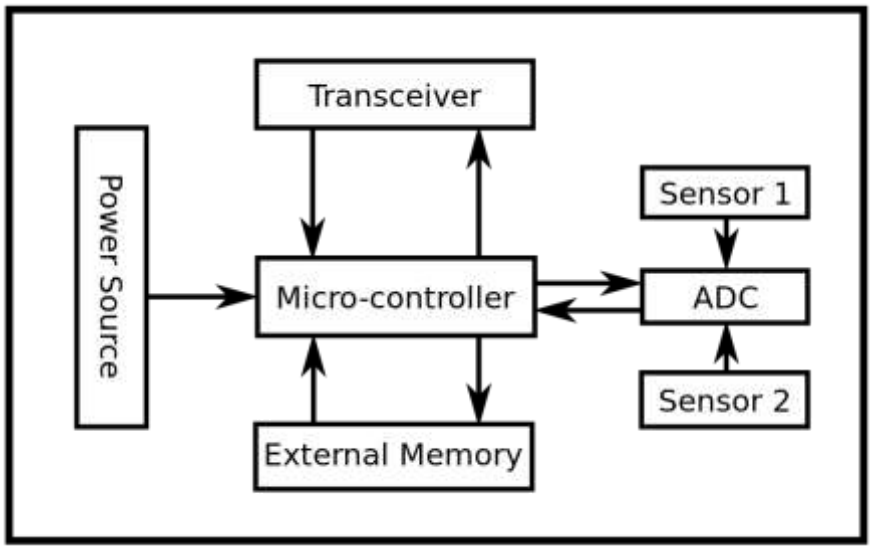
VIII. Capability to ensure strict environmental conditions

Sansor node:-

Sansor nodes are the main compants of wireless sensor network.Sensor node is capable of sensing,gathering and processing data and communicating with other connected nodes in the network. The main function of sensor node is to sense data and sends it to the base station in multi hub enviromenta for which routing path is essential, For computing the routing path from the source node to base station ther is huge numbers of proposed routing also and routing protocol exist.

WIRE LESS SENSOR NODE ARCHITECTURES:-

There are main five components of a sensor node.They are microcontroller, Transceiver ,extranal memory , power source and one or more sensors.



Function of Sensor node in Industry:-

Mechanical plants incorporate sensors associated with the control station through wire and remote techniques for persistent detecting and checking the status of the framework. Remote innovation gives an appropriate help to the business . It offers numerous points of interest like low establishment cost, adaptability, adaptability, absence of cabling, clever handling capacity, high versatility and simplicity of arrangement contrasted with customary wired arrangements. These focal points are promising for industry, where a significant development sooner rather than later is normal.

Shortest Path Routing Algorithm:-

- ➢ This algorithm find minimum distance between source node to destination node.
- ➢ Selects shortest distance after comparing with others.
- ➢ Forward packet to the next router with minimum distance.
- ➢ A path between source to destination with minimum distance is selected.

Dijkstra's shortest path algorithm in Route computation;-

The issue of finding most brief way assumes a significant job in the plan and examination of system. Most routing can take care of as shortest way issue once and suitable cost is appointed to each connection, mirroring its accessible band width, deferral or bit mistake proportion. This algorithm is utilized for processing the briefest way from root hub to each orther hub in the network.It is fundamentally the same as the the prim's algorithm for minimum spanning tree..

Problem definition -

For the moment we may accept the model as, to identifying the yarn break in a turning machine in a material industry is the significant task as it straightforwardly influences the creation efficiency.The end breakage in ring turning not just diminishes the running productivity of the procedure yet in addition diminishes the nature of the yarn .With the speed up the yarn breakage rate increases.The above spatial issue emerges because of a higher speed causes amazing radiating powers on the strands bringing about poor traversing strength and an expansion in yarn tension,which causes high yarn breakage.Hence to defeat from the above problem,an proper sensor organize is constantly required to screen the framework conduct so that reasonable sign preparing and damage sensitive feature extraction on the deliberate information can be performed efficiently.The number of detecting frameworks accessible varies essentially relying on the activity.Current detecting system in turning machine utilizes wired concept.Wireless correspondence can cure the cabling issues with the conventional checking framework and altogether lessen the upkeep costs.

Algorithm-

ALGORITHM USED FOR ROUTING

To get our objective we use two routing algorithm.

1. Prim's Algorithm

2. Dijkstra's algorithm

Prim's Algorithm:-

With the help of Prim's Algorithm we try to find out the minimum spanning tree from a weighted graph. So a subset of edges is found which forms a tree which consists of every vertex where there is a minimum value of the summation of weight of the edges. If the graph is having an isolated vertex then it will find out the minimum spanning tree of the connected vertex.

Pseudo code:

Minimum-Spanning-Tree-by-Prim(G, weight-function, source)

1 **for each** vertex *u* **in** graph **G**

2 **set** key **of** *u* **to** _

3 **set** parent **of** *u* **to** *nil*

4 **set** key **of** source vertex **to** *zero*

5 **enqueue to** minimum-heap **Q** all vertices in graph **G**.

6 **while Q** is not *empty*

7 **extract** vertex *u* from **Q**

8 **for each** adjacent vertex *v* **of** *u* **do**

9 **if** (*v* is *still* in **Q**) and (weight-function(u, v) < key of *v*)

then

10 **set** *u* **to** be parent **of** *v // in minimum-spanning-tree*

11 update *v*'s key to equal weight-function(u, v)

Dijkstra's algorithm:-

Dijkstra's algorithm also known as the single-source shortest path. Through this algorithm we get the length of the shortest path from the one vertex which can be considered as source to each of the remaining vertices in the graph.

Dijkstra's –A Greedy Algorithm

Dijkstra's algorithm uses the greedy approach to solve the single source shortest problem. In this algorithm an unselected vertex is selected, vertex v nearest to the sources and declares the distance to be the actual shortest distance from s to v. The edges of v are then checked to see if their destination can be reached by v followed by the relevant outgoing edges.

Pseudo code:

1 **function** Dijkstra(*Graph*, *source*):

2 **for each** vertex *v* in *Graph*: *// Initializations*

3 dist[*v*] := infinity ; *// Unknown distance function from*

4 *// source to v*

5 previous[*v*] := undefined ; *// Previous node in optimal path*

6 **end for** *// from source*

7 dist[*source*] := 0 ; *// Distance from source to source*

9 *Q* := the set of all nodes in *Graph* ; *// All nodes in the graph are*

10 *// unoptimized - thus are in Q*

11 **while** *Q* **is not** empty: *// The main loop*

12 *u* := vertex in *Q* with smallest distance in dist[] ; *// Start node in first case*

13 remove *u* from *Q* ;

14 **if** dist[*u*] = infinity:

15 **break** ; *// all remaining vertices are*

16 **end if** *// inaccessible from source*

17 **for each** neighbor *v* of *u*: *// where v has not yet been*

19 *// removed from Q.*

20 *alt* := dist[*u*] + dist_between(*u*, *v*) ;

21 **if** *alt* < dist[*v*]: *// Relax (u,v,a)*

22 dist[*v*] := *alt* ;

23 previous[*v*] := *u* ;

24 decrease-key *v* in *Q*; *// Reorder v in the Queue*

25 **end if**

26 **end for**

27 **end while**

28 **return** dist;

If we are only interested in a shortest path between vertices *source* and *target*, we can

terminate the search at line 13 if *u* = *target*. Now we can read the shortest path from *source* to

target by reverse iteration:

1 *S* := empty sequence

2 *u* := *target*

3 **while** previous[*u*] is defined: *// Construct the shortest path with a*

stack S

4 insert *u* at the beginning of *S // Push the vertex into the stack*

5 *u* := previous[*u*] *// Traverse from target to source*

6 **end while** ;

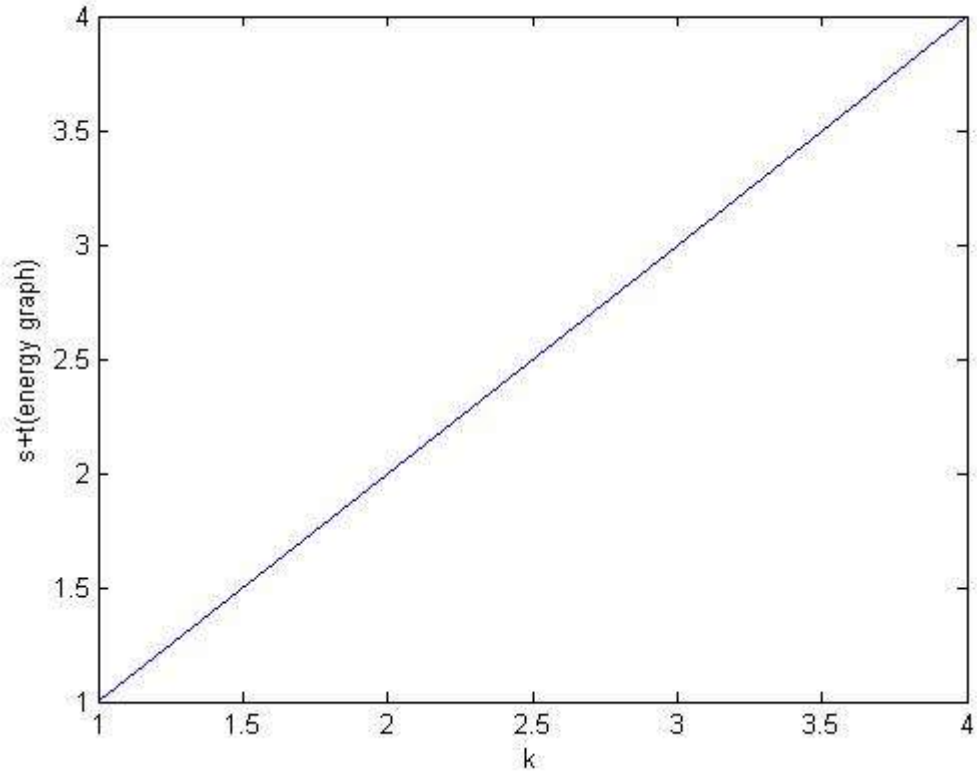
RESULT ANALYSIS USING MATLAB:

As we know that the MATLAB is a programming language which is basically used for research work and to compute the logic in terms of mathematically, Hence if we want to first compute the result of our thesis than we have to first write the program. Here I had used an Dijkestra algorithms as well as Prims Algorithms to compute my problem. We know that both these algorithms are used to find the shortest path and both are effective but still some differences are there. Basically these two algorithms are looking same but the major differences are:

1. Prim's algorithm stores a minimum cost edge whereas Dijkstra's algorithm stores the total cost from a source node to the current node.
2. Dijkstra's gives you a way from the source node to the destination node such that the cost is minimum. However Prim's algorithm gives you a minimum spanning tree such that all nodes are connected and the total cost is minimum.
3. Dijkstra's algorithm finds the minimum distance **from node i to all nodes** (you specify i). So in return you get the minimum distance tree from node i.
4. Prims algorithm gets you the minimum spaning tree **for a given graph**. A tree that connects all nodes while the sum of all costs is the minimum possible.
5. So with Dijkstra's **you can go from the selected node to any other with the minimum cost**, you don't get this with Prim's
6. Dijkstra's tries to find the shortest path between it's beginning node and each other node. So it actually wants to reach every other node as efficiently as possible.
7. Prim also tries to find the shortest path, but this time we don't keep comparing with the beginning node. The comparison does continue, but only for other nodes among each other.

Experimental Results:

By implementing the above algorithms here in MATLAB we had obtained the following graph, which is as showed bellow.



NOTE:

1. Here in the graph showed above the X-axis is for the distance of n-Nodes.
2. Here in the graph showed above the Y-axis is for the distance of n-Nodes with respect to time interval t.
3. According to law of Conservation and Newton's the rate of change of momentum is nothing but the Power or Energy.
4. When we are moving to X-axis w.r.t Y-axis in interval t than the subsequent amount of energy is either generated or liberated from the node.

5. Here in our case the energy is consumed by the node while travelling to a particular target.
6. Hence the experimental analysis result is as tabulated bellow which shows that in each of the case how much total path has been covered and how much energy is being consumed by the node to travel to specific target node.

Here we are going to take three comparison cases which are tabulated as follows.

Sl No	CASE	Dijkestra No of Nodes	Path Covered	Energy Consumed	Prims No of Nodes	Path Covered	Energy Consumed	Energy Saved
1	1	3(A, B,C)	BAC	184.0027	3(A,B,C)	BA	184.0028	0.0001 J/S
2	2	4(A,B,C,D)	ABCD	189.0029	4(A,B.C,D)	BA	190.0021	0.9992 J/S
3	3	6(A-D)	A-D	220.0213	6(A-D)	ABDC	226.0213	6.0 J/S

So the above points are useful in the sense when we are considering the distance factor. Again we know that the energy is also another important factor for us. So while calculating the energy factor we know that as we move more & more, maximum is the energy wasted.

Hence as per the movement the prims consumes maximum energy where as Dijkstra's takes less energy to do the same job.

Hence from the above tabulation it is conclude that, Both these algorithm are used for finding the shortest path. But in prims algorithms the energy consumption is maximum than dijkestra and path coverage is less.

Future Work:

In contrast to the above said case we may enhance our paper to the future work like,

1. To calculate the ratio of energy consumption by each node in each of the algorithms.

2. We may enhance our thesis in terms of calculating how to make the industry fully automated.

REFERENCES

1. I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor

   networks," *Communications Magazine,IEEE*, vol. 40, pp. 102-114, 2002.
2. 2 J. L.Hill, "System architecture for wireless sensor networks," University of California,Berkeley, Ph.D. dissertation 2003.
3. IEEE Standard 802.15.4, "Wireless Medium Access Control (MAC) and Physical Layer(PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)," 2006.
4. Brian Otis, Jan Rabaey, *Ultra-Low Power Wireless Technologies for Sensor Networks*.:Springer, 2007.
5. Jennifer Yick, Biswanath Mukherjee, Dipak Ghosal, "Wireless sensor networksurvey," *Computer Networks,Elsevier*, vol. 52, pp. 2292-2330, 2008.
6. G. Simon, M. Maroti, "Sensor network-based counter sniper system," in
   a. *Proceedings of the 2nd international conference on Embedded networked sensor*
   b. *systems*, Baltimore, MD, 2004, pp. 1-12.