IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 3, March 2016

A Survey on Node Scheduling Methods for Wireless Sensor Networks

Neha Patel[1], Shweta Singh[2]

PG Student, Department of Computer Engineering, CGPIT, Bardoli, Gujarat, India[1]

Assistant Professor, Department of Computer Engineering, CGPIT, Bardoli, Gujarat, India[2]

Abstract: The wireless sensor networks (WSN) is a combination of a large number of low-power, short-lived, unreliable sensors. The main challenge of wireless sensor network is to obtain long system lifetime. Many node scheduling algorithms are used to solve this problem. This methods can be divided into the following two major categories: first is round-based node scheduling and second is group-based node scheduling. In this paper many node scheduling algorithm like AFAP, RSGC, Tree-Based distributed wake-up scheduling and Clique based node scheduling Algorithm are analysed.

Keywords: WSN(The wireless sensor networks); AFAP(As-Fast-As Possible); RSGC(A randomized node scheduling); Clique;

I. INTRODUCTION

In the wireless sensor network many tiny sensing devices are deployed in a region of interest. Each device has processing and wireless communication capabilities, which used to collect information from the environment and then it will generate and deliver report messages to the remote base station (remote user). The base station collects and analyses the report messages received and decides whether there is an unusual or concerned event occurrence in the deployed area[1]. Considering the incomplete capabilities and susceptible nature of an individual sensor, a wireless sensor network has a large number of sensors deployed in high density. Thus redundancy must be broken to increase data accuracy and sensing reliability. Usually battery power Energy source is provided for sensors, which has not yet reached the stage for sensors to run for a long time without recharging in wireless sensor networks. Moreover, since sensors are often anticipated to work in remote or aggressive environment, such as a battlefield or desert, it is unwanted or impossible to recharge or replace of all the sensors' battery power. Long system lifetime is anticipated by many monitoring applications. The system lifetime, it means the time until all nodes have been drained out of their battery power or the network no longer provides an acceptable event discovery ratio, directly affects network usefulness. Therefore, energy efficient design for extending system lifetime without surrendering system original performances is an important challenge to the design of a large wireless sensor network.

All nodes share common sensing tasks, which suggests that not all sensors are required to perform the sensing tasks during the whole system lifetime in wireless sensor networks[3]. Some nodes sleep condition does not affect the overall system function providing there are enough working nodes to assure it. Therefore, if we firstly deploy a large number of sensors and schedule them to work simultaneously, system lifetime can be extended constantly it means redundancy is used to increase system lifetime. Many node scheduling algorithms are used to solve this problem. This methods can be divided into the following two major categories: first is round-based node scheduling and second is group-based node scheduling. The sensor nodes will perform the scheduling algorithm during the initialization of each round in round-based node scheduling method. This kind of methods requires each sensor node to execute the scheduling algorithm for more than once during its lifecycle. In a group-based node scheduling method, each node will perform the scheduling algorithm only once after its deployment[7]. All sensor nodes will be distributed into some different groups after the execution of the scheduling algorithm. After that in each of the followed time slots, each group of nodes will keep active in turn.

II. LITERATURE SURVEY

A. *AFAP Scheduling Algorithm*

AFAP is known as As-fast-As-possible scheduling algorithm.

```
entity prefetch is
  port (branchpc, ibus  : in bit32;
        branch, ire     : in bit;
        ppc, popc, obus: out bit32);
end prefetch;

architecture behavior of prefetch is
begin
  process
   variable pc,oldpc : bit32 := 0;
  begin
   ppc  <= pc;              --1
   popc <= oldpc;           --2
   obus <= ibus + 4;        --3
   if (branch = '1')        --4
    then
     pc := branchpc;        --5
   end if;                  --6
   wait until (ire = '1');  --7
   oldpc := pc;             --8
   pc    := pc + 4;         --9,10
  end process;
end behavior;
```

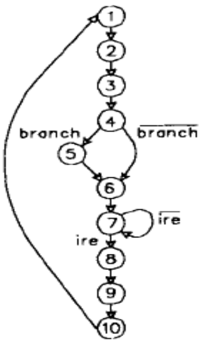

Figure 2.1 Behavioural description example[2].

A interactive description of the problem to schedule is given by the control-flow directed graph B = (V, E). The nodes v represent operations to be scheduled, and the

DOI 10.17148/IJARCCE.2016.53225

edges give the priority relation. Vi is an instant predecessor (called just predecessor) of Vj. Vj is called an instant successor (or just successor) of Vi. The explanation of B is imperative. An operation is executed after one of its predecessors is executed.

Fig. gives an example of the control-flow graph. Nodes 1 and 2 agree to signal assignments. Nodes 4 and **7** are restricted branches, their outgoing edges are labeled with the consistent conditions. Nodes 5, 8 and 10 are variable assignments. Nodes 3 and 9 are additions. Node 6, finally, is a dummy node corresponding to the "end if" statement. The program is assumed to loop boundlessly, i.e., node 1 executes after node 10. The control-flow graph has a unique first operation V1, at which execution starts; in the example this is node 1. It should be possible to reach all other operations from V1**,** otherwise there are dead operations in B that can never be executed.

The longest path of the control-flow graph is a path starting at V1 and ending at an operation with no successors. Repetition of operations are not measured, i.e... Cycles in the graph are navigated just once for longest path calculation. The set of all longest paths is denoted as {pi*}*. It represents all different operation sequences (again, excluding repetition of cycles) that the quantified behavior allows.

The AFAP scheduling problem is then formulated as follows. Given ***B*** = (V, E) and a set of constrictions, schedule all operations ***v*** such that all possible longest paths {pi} execute in the lowest number of control states and all constraints are met.

The algorithm for AFAP scheduling is following steps:
1) Converting the control-flow graph *E* into a directed acyclic graph (DAG) and keeping lists for the loops.
2) All paths in the DAG are organized AFAP autonomously, according to the data-flow constraints in each path.
3) The schedules of step *2)* are covered in a way that minimizes the number of control states.
4) The finite state machine for control is built[2].

B. The random scheduling algorithm RSGC

A randomized node scheduling method confirms the coverage quality and network connectivity instantaneously and the steps for the method are:

Step 1: Select a sub set randomly: Initially, each sensor node A produces a random number ai between 0 to k-1 and assigns itself to subset ai.

Step 2: Broadcast minimum hop count: The step is to broadcast a HOP advertisement message to its instant neighboring sensor nodes from the sink node at the time. Each HOP advertisement message encompasses the minimum hop count to the sink, the node ID and its sub set decision. The minimum hop count is set to 0 in the packet broadcast from the sink. Initially, the minimum hop count to the sink is set to infinity at each sensor node. After receiving a HOP advertisement message, each node will put the message in its buffer. It will submit the broadcast of the HOP message after a back-off time and on lyre-broadcasts the HOP message that has the minimum hop count. Before there- broadcast of the HOP message, the hop count value in the HOP message is increased by1. HOP message broadcasts with an on-minimal hop count will be repressed if the HOP message with the actual minimal hop count arrives before the back-off time expires with this method. The number of broadcasts from each sensor node depends on the length of back-off time. When the back-off time increases it will meaningfully decrease the number of broadcasts.

Step 3: Exchanging information with local neighbors: the sensor node broadcasts information consists of its minimum hop count, its node ID, its subset decision, the node ID so turns up stream nodes and their subset decisions. The current node receives its minimum hop count from the nodes which are known as the upstream nodes. Each sensor node contains and maintains all the information it receives from its near nodes.

Step 4: Enforce the extra-on rule: Each sensor node selects the extra time slots based on the extra-on rule and the information from Step3 it has to continue active to guarantee network connectivity and updates its working schedule accordingly. Then the updated working schedule is broadcasted to neighboring sensor nodes. If a sensor node A has a downstream node B, which is active in time slot i, and if none of node B's upstream nodes is active in that time slot, then node A should also work in time slot this is known as extra-on rule. In other words, also working in the duty cycles allocated by the randomized coverage-based scheduling, node A is required to work in extra time slots, e.g., time slot in this case.

Step 5: Work according to the new working schedule[3].

C. *Tree-Based distributed wake-up scheduling*

Different from the above node scheduling method, around-based node scheduling method for wireless sensor networks(WSN) called Tree-Based distributed wake-up scheduling is proposed in Wu and Tseng(2009), in which, AWSN is modeled as a nun directed graph G¼(V;E), where V contains all nodes and E contains all communication links between nodes. A set of nodes RDV is requested to conduct data collection and each node needs to periodically send its sensed data to the sink, and these data may be combined on their way to the sink. The goal of the Tree-Based scheduling scheme is to construct a sub tree T from G rooted at the sink connecting all nodes in R and schedule the wake-up time of nodes in T for energy-saving and low-latency purposes.

In the Tree-Based scheduling scheme it is a time-division model is accepted by dividing time into fixed-length slots. Each k following slots are grouped together and called a frame. In each frame, each node vi in T will be assigned a wake-up slot si Af0; 1; . . . ; k_1g. During slot si, vi must wakeup to announce a beacon to synchronize with its children and then collect sensed data from them. Without si, vi may go to sleep.

Two phases of the Tree-Based scheduling scheme are:

Step 1: Tree-formation phase: In this phase a sub tree T is build and all nodes are connected in R.

Step 2: Slot-allocation phase: In this phase it is to find an interference-free slot for each node in T with low latency LT, where the interference set of vi with respect to T for a given node vi and a data collection tree T in G is defined as It(Vi)=N(Vi)UN(Ct(Vi))UPt(N(Vi))-{Vi}.

Here, Pt(Vi) denotes the set of Vi's parent in t, Ct(Vi) denotes the set of Vi's children in t, N(Vi) denotes the set of Vi's neighbors of G[4].

D. *Clique based node scheduling Algorithm*

First algorithm is for Distributed maintenance of connectivity for each group:

For any sensor node pi, let node pj be a neighboring node of pi. Suppose the minimum hops to the Sink from node pi, pj are Hi and Hj respectively. If Hi¼Hj+1, then node pj is called an Upstream Node of pi and the node pi is called a Downstream Node of pj. If Hi¼Hj, then node pj is called a Brother Node of pi. If pj>pi, then node pj is called an Older Brother Node of pi, otherwise, node pj is called a Younger Brother Node of pi.

Algorithm (CMEG: Connectivity Maintenance for Each Group)[1]

If a sensor network is connected, and each node conserves a list of all Upstream Nodes that are on its shortest paths to the Sink and all of its Brother Nodes. Each node in the sensor network will update its GID List to maintain the connectivity of each group of the network after the GIDs assignment.

Step 1: For a node pi, suppose that its GIDs Listi s Li. For any g belongs to Li:

Step 1.1: If there is neither a Upstream Node n or a Brother Node, which has g in its GID List, then pi selects one who has the shortest GIDs List of its Upstream Nodes, pj, and sends an AGAC (Appended GID Application for Connectivity) message to pj.

Step 1.2: If there is no Upstream Node but there are nodes who have g in their GID List that is Brother Nodes. Then pi selects one of its Brother Nodes, pj, who has the shortest GIDs List, and sends an AGAC (Appended GID Application for Connectivity) message to pj.

Step 2: Each node in the sensor network keeps a back-off timer after the GIDs assignment.

Step 2.1: If the node pj received AGAC messages from it's any Downstream Node or Younger Brother Node pi before timeout. Then it will update its GID List according to the AGAC messages. Step 2.2: If node pj finds out that it has no Upstream Node and the node pj received AGAC messages from its any Older Brother Node pi before timeout that has g in its GID List, then it will select one of its Upstream Nodes px, who has the shortest GIDs List, and sends an AGAC (Appended GID Application for Connectivity) message to px.

Step 2.3: It will broadcast its updated GIDs List to all of its neighbors after timeout.

Second algorithm is for Connectivity and coverage maintenance scheduling algorithm.

Suppose that a sensor network is m-covered and connected. Then all nodes in the sensor network can be scheduled into k different groups {0; 1; . . . ; k-1}; k<=m, using the following steps:

Phase (1): GIDs Assignment

Step 1: Every sensor node pi contains an information table ITi which includes data like IDNN which is known as ID of its Neighboring Nodes, NNLN it means Neighboring Nodes List of its Neighbors, IUN known as Identification of Upstream Node, TSHS means The Shortest Hops to the Sink, and the parameter k. At the first stage, ITi is an empty table. After that the Sink broadcasts a Hello message and the parameter k. Each node that received this message record k will set TSHS as 1.

Step 2: Every node with non-empty TSHS propagates a Hello message including its ID, k and TSHS. If any node pi it received a message with the TSHS j and parameter k from node pj, the n it compares the TSHS j with the records TSHS i in its ITi. If the TSHS i is empty, then it records the ID of pj (in IUN), the TSHS (¼the receivedvalueplus1) and parameter k into ITi. It checks whether TSHSi¼TSHSj + 1 if the TSHS is not empty. pj is recorded in IUN if the equation is true. pi will also updates IDNN in ITi when it receives messages.

Step 3: Each node generates and broadcasts a message including ID and IDNN. Each node confirms its active neighboring nodes through information exchanges among neighboring nodes and forms its NNLN in the table.

Step 4: Run the Algorithm1 to assign GIDs.

Phase (2) Connectivity Maintenance

Step 5: Run the Algorithm to update the connectivity.

Phase (3) Group Working

Step 6: All nodes work in turns at their given time slots in the working phase. At the time slot t, if t=g mod k, then all nodes in group g keep working, while other nodes will hibernate.

III.COMPARATIVE ANALYSIS

Table 3.1. Comparative analysis of node scheduling methods

Authors	Method	Description
Rad Camposano [2]	AFAP(As-fast-As Possible) Scheduling Algorithm .	Contains minimum number of control steps, taking into account arbitrary constraints that limit the amount of operations in each control step. The result is a finite state machine that implements the control.
Lei Wang[1], Liu C, Wu K, Xiao Y and Sun B[3]	The random scheduling algorithm RSGC	A randomized node scheduling method ensures the coverage quality and network connectivity simultaneously.

Lei Wang [1], Wu FJ and Tseng YC[4]	Tree-Based distributed wake-up scheduling	A round-based node scheduling scheme for wireless sensor networks used for energy- saving and low-latency purposes.
Lei Wang, Ruizhong Wei, Yaping Lin and Bo Wang[1]	Clique based node scheduling Algorithm	Solve the node scheduling problem for m-covered and connected sensor networks.

IV. SIMULATION SCENARIO

Simulation parameters:

Table 4.1: Simulation parameters

Parameters	Values
Simulator	NS 2.34
Number of nodes	8
Area size	100X100 m^2
Routing protocol	AODV
Simulation time	100ms

Implementation scenario:

The scenario using scheduling and without scheduling is implemented. Here 8 nodes are taken for wireless communication.

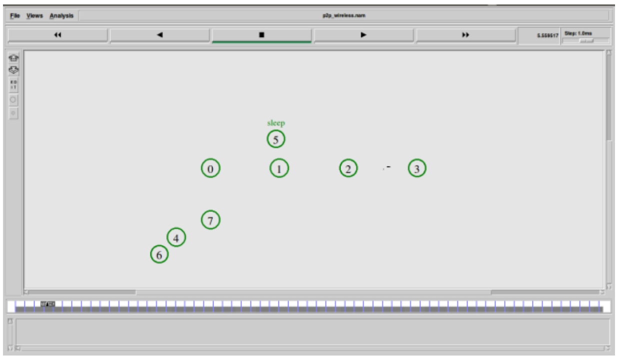

Figure 4.1: Implementation scenario

The AODV protocol is used for data transmission in the area of 1000m X 1000m. Here simple scheduling is used for simulation where node 5 is in sleep state while transmission. The results for scheduling and without scheduling is presents.

Simulation Results:

The simulation results are given below.

Table 4.2: Results

Parameters	Normal scenario	With scheduling
Average remaining energy	111.523159 joule	113.910986 joule
Average throughput	192.26kbps	212.92kbps
Average End-to-End Delay	274.144 ms	275.207 ms
Packet delivery ratio	0.7099	0.9059

The results are given here with parameters Average remaining energy, Average throughput, and Average End-to-End Delay and Packet delivery ratio.

V. CONCLUSION

Different node scheduling algorithms are survey in this paper. All the methods have different ability to solve different problems. The wireless sensor networks biggest issue is network lifetime which is solved by this node scheduling algorithms. From all the algorithm the clique based node scheduling method that is group based node scheduling method which includes location information guarantee that each group will be still connected and maintain the coverage ratio as high as possible. So clique based node scheduling algorithm is an efficient method for wireless sensor networks. From the results we can say that using scheduling is an effective way to get more lifetime of the network.

ACKNOWLEDGMENT

It is a genius pleasure to express my deep sense of thanks and gratitude to my mentor and guide **Ms. Shweta Singh** for his timely advice; scholarly advice and scientific approach have helped me to a very great extent to accomplish this study. Also I would like to express my gratitude for her continuous motivation and her suggestions.

REFERENCES

1. Lei Wang, Ruizhong Wei, Yaping Lin and Bo Wang, "A clique base node scheduling method for wireless sensor networks", *ELSEVIER Journal of Network and Computer Applications* 33 (2010) 383–396, 2010
2. Rad Camposano. "Path-Based Scheduling for Synthesis", *IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN*. VOL. IO. NO. I. JANUARY 1991
3. Liu C, Wu K, Xiao Y, Sun B. Random coverage with guaranteed connectivity: joint scheduling for wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems* 2006; 17(6):562–75.
4. Wu FJ, Tseng YC. Distributed wake-up scheduling for data collection in tree-based wireless sensor networks. *IEEE Communications Letters* 2009; 13(11):850–2.
5. Hicham Lakhlef , Abdelhak Bentaleb, Jean-Frederic Myoupo, Abdelhak Boubetra and Saad Harous, "Cliques and Clusters based Hierarchical Scheme for Sensor Networks Partitioning**", ***IEEE GCC Conference and exhibition***, November 17-20,2013
6. Sumit Kumar and Siddhartha Chauhan, "A Survey on Scheduling Algorithms for Wireless Sensor Networks", *International Journal of Computer Applications (0975 – 8887) Volume 20– No.5, April 2011*
7. Fang-Jing Wu and Yu-Chee Tseng "Distributed Wake-up Scheduling for Data Collection in Tree-Based Wireless Sensor Networks", *IEEE COMMUNICATIO"NS LETTERS*, VOL. 13, NO. 11, NOVEMBER 2009
8. Yanwei wu, Xiang-Yang Li, YunHao Liu and Wei Lou, "Energy efficient wake up scheduling for data collection and aggregation", *IEEE Transactions on Parallel and Distributed systems*, vol 21, No. 2 February 2010
9. Akyildiz IF, Su W, Sankarasubramaniam Y, Cayirci E, "A survey on sensor networks", *IEEE Communications Magazine* 2002;40(8):102–14.
10. TSAI YR, "Coverage preserving routing protocols for randomly distributed wireless sensor networks", *IEEE Transactions on Wireless Communications* 2007;6(4): 1240–5.
11. Vaseashta A, Vaseashta S., "A survey of sensor network security" *Sensors & Transducers Journal* 2008;94(7):91–102.

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 3, March 2016

12. Yong D, Chen W, Li X, "A connectivity based partition approach for node scheduling in sensor networks" *In: Lecture notes in computer science*, vol. 4549, 2007. p. 354–67

BIOGRAPHIES

Neha Patel has received his B.TECH. Degree in Computer Engineering from Sardar Vallabhbhai Patel institute of technology, Vasad, Anand, Gujarat-2014. Currently she is pursuing M.TECH degree in Computer Engineering at Computer Engineering at CGPIT, Bardoli - (Uka Tarsadia University), Dist. Surat, Gujarat, India. Her areas of interest are Wireless Sensor Network, Wireless Communication and Mobile Computing, Data mining and Image Processing.

Guided by: **Shweta Singh** received her M.TECH. Degree from Jaypee University of Information Technology, Waknaghat, Distt. Solan-173215.Currently she is working as an Assistant Professor in Computer Engineering at CGPIT, Bardoli - (Uka Tarsadia University), Dist. Surat, Gujarat, India. Her areas of interest are Wireless Sensor Network, Wireless Communication and Mobile Computing and Advance Computer Network.