Adaptive Sleep Wakeup Scheduling Technique for WSN

[1]Sasikala V, [2] Dr.C. Chandrasekar
[1]DRBCCC Hindu College, Chennai, Tamil Nadu
[2]Dept. of CSE., Periyar University, Salem, TamilNadu, India
[1]vsasikalaphd@gmail.com

Abstract—

Wireless sensor network (WSN) is an emerging technology in the today's world. The WSN has been a prominent technology in many fields so it is very important to decrease the energy consumption by the network and increase the network's life. To achieve this we propose an adaptive Sleep/Wake scheduling technique for WSN. In this approach, we initially consider some nodes as initiator nodes through these nodes we decide the cluster head. Then these cluster head forms clusters in the network. In these clusters, the cluster head selects a node with the highest energy, keep it in active mode, and sends the remaining nodes into the sleep mode and a traffic adaptive sleep-wakeup scheduling technique is provided. simulation results, we show that the proposed scheduling technique increases the average residual energy and throughput.

I. INTRODUCTION

I.1 WSN

A WSN is structured with a certain number of tiny sensor devices, and each device has the abilities of computation, storage, and communication. This enables it to collect sensing data and conduct data processing tasks about the environment, and to generate and deliver helpful information on the monitored objects to the base station for decision making. The appearance of WSNs has considerably changed various kinds of remote sensing applications such as environmental and ecological monitoring of natural habitats, smart homes, and military areas.

Wireless sensors are usually densely deployed on the target field. By this many sensors can detect an event, deliver and receive the sensed data packets simultaneously. Due to the facts that wireless sensors are physically small and must use extremely limited power or energy, the network lifetime is an essential consideration in sensor network applications. Moreover, a WSN is usually deployed in hostile fields or under harsh environments where manually recharging batteries for sensors is not feasible, one typical alternative approach to energy saving is to turn off some sensors and activate only a necessary set of sensors while providing a good sensing coverage and network connectivity simultaneously. A good sleep wake scheduling has to provide an even distribution of energy consumption among sensor nodes so that the network lifetime is extended [1] [7].

I.2 Power aware scheduling in WSN

Wireless sensor networks are battery powered, therefore prolonging the network lifetime through a power aware node organization is highly desirable. An efficient method for energy saving is to schedule the sensor node activity such that every sensor alternates between sleep and active state. One solution is to organize the sensor nodes in disjoint covers, such that every cover completely monitors all the targets. These covers are activated in turn, in a round-robin fashion, such that at a specific time only one sensor set is responsible for sensing the targets, while all other sensors are in a low-energy, sleep state.

Scheduling nodes to enter low energy states is efficient way to accomplish energy savings. An effective way to conserve energy is to schedule apriori the wireless node transmissions, allowing them to enter a low state energy while they are inactive. Energy saving fall in one of the following categories: (1) schedule operations, to allow nodes to enter low energy states; (2) choose routes that consumes the lowest energy; (3) selectively use wireless nodes based on their energy status; (4) reduce amount of data and avoid useless activity [2] [3] [6].

I.3 Duty-cycle adjustment in WSN

In wireless sensor networks, the major sources of energy consumption are due to idle listening, control packet overhead, collision and overhearing. To improve the network lifetime nodes are put into sleep mode periodically. This corresponds to introducing a duty cycle, interleaving sleep and active periods at the device level. Also to reduce the energy consumption of idle listening, duty cycle operation is introduced in S-MAC. Each sensor node in S-MAC follows a periodical synchronized listen-sleep schedule and operates at low duty cycle to conserve energy. T-MAC, a variation of S-MAC adopts timeout and adaptive duty cycle method to improve energy efficiency. A-MAC introduces an adaptive duty cycle mechanism, which let length of cycle increases in an exponential style according to traffic for energy conservation and latency reduction requirements. However, the length of cycle can not change freely according the traffic and the efficiency of energy conservation and latency reduction is limited. Furthermore, maintaining synchronization between nodes leads to energy waste and time cost [9].

The use of duty cycle algorithms requires periodic beacons in order to update the nodes neighbor tables. Adaptive duty-cycling has been proposed in the recent works on energy-harvesting technologies such as solar power, to replenish battery supply in WSNs. Due to high costs and the unavailability of a continuous power supply, it is not feasible to have instantly sufficient energy output. Hence, saving idle energy consumption is still necessary. Adaptive duty cycling is thus proposed to save energy consumption and to prolong the sustainable workable time per node. The duty cycle setting can be based on the residual energy, node location, or the rechargeable energy on each node, independently [10] [11].

I.4 Problem Identification

In our previous work, we have proposed a technique which combines energy efficiency and multiple path selection for data fusion in WSN. The network is partitioned into various clusters and the node with highest residual energy is selected as the cluster head. The sink computes multiple paths to each cluster head for data transmission. The distributed source coding and the lifting scheme wavelet transform are used for compressing the data at the CH. During each round of transmission, the path is changed in a round robin manner, to conserve the energy. This process is repeated for each cluster.

The main drawback of this approach is that it consumes lot of network’s energy since when the transmission takes place in the network the all the nodes in the network stay in active mode without transmitting anything. This leads to the decreasing of network’s lifetime. To overcome this drawback we propose a guaranteed distributed sleep/wake scheduling scheme, where only the transmitting node stay in active mode and the other nodes goes to the sleep mode.

II. LITERATURE REVIEW

Soumya Ray et al [4] have proposed on energy efficient token based MAC protocol integrated with sleep scheduling for WSNs, in order to reduce energy consumption of each sensor node. As it is one of the important issues to prolong the network lifetime. To derogate energy consumption most of the MAC protocols in WSN exploits low duty-cycle; among those RMAC, HEMAC allows a node to transmit data packet for multi-hop in a single duty-cycle. To reduce energy consumption on prolonged network life time sensor networks are usually duty cycled; each node remains in low power sleep mode most of the time and wakes up periodically to sense for channel activities. In the above said MAC protocols, due to the synchronized scheduling and transmission collisions, flooding increases resulting in energy waste and low throughput. Allowing for nodes to operate with a new sleep based token approach; they intend to produce energy efficiency in an event based approach by cutting down flooding, collision and traffic congestion. However as the traffic load increases there are a slight increase in the energy consumption.

Giuseppe Anastasi et al [5] have proposed an Adaptive Staggered sLEEp Protocol (ASLEEP) for efficient power management in wireless sensor networks targeted to periodic data acquisition. This protocol dynamically adjusts the sleep schedules of nodes to match the network demands, even in time-varying operating conditions. In addition, it does not require any a-priori knowledge of the network topology or traffic pattern. The results obtained show that, under stationary conditions, the protocol effectively reduces the energy consumption of sensor nodes (by dynamically adjusting their duty-cycle to current needs) thus increasing significantly the network lifetime. With respect to similar non-adaptive solutions, it also reduces the average message latency and may increase the delivery ratio. Under timevarying conditions the protocol is able to adapt the duty-cycle of single nodes to the new operating conditions while keeping a consistent sleep schedule among sensor nodes. In addition, its robustness against communication errors could be further enhanced by means of traditional techniques for increasing wireless reliability.

Rodolfo de Paz and Dirk Pesch in [8] have proposed a duty-cycle learning algorithm (DCLA) to enhance the overall network performance of IEEE802.15.4 beacon-enabled networks. They employ a reinforcement learning algorithm to solve the problem of adapting the coordinator's duty-cycle. It is made according to end devices traffic conditions with the objective of minimizing the energy consumption while balancing at the same time other important WSN performance parameters. The results show that the DCLA outperforms current state of the art on duty-cycle adaptations for IEEE 802.15.4 beacon-enabled networks in terms of average drop rate, throughput, energy efficiency and end-to-end delay. However it consumes much more energy as it cannot sleep even the traffic in the network is reduced being thus it is less energy efficient.

LIU Hao et al in [9] have proposed an adaptive MAC (CBA-MAC) protocol for clustering based wireless sensor networks. In CBA-MAC, sensor nodes are divided into cluster head node and normal node. Cluster head node collects traffic information from normal nodes and calculates appropriate duty cycle according to current traffic, then broadcasts duty cycle information to normal nodes, so sensor nodes can fit for variable traffic occasions well. Additionally, a proper sleep scheme is also proposed for energy saving consideration. They have implemented CBAMAC in ns-2, and the simulation results shows that the energy consumption and latency is reduced well. However sensor nodes can only adjust the energy consumption according to the traffic.

Rone Ilidio da Silva et al in [11] have proposed an energy-efficient in-network spatial query processing mechanism that assumes nodes having no knowledge about their neighbors. The spatial query process was divided in stages. For the Forwarding and Return stages, they have proposed a new location-based routing protocol called ABF, which uses only active nodes to forward packets. In the Dissemination and Aggregation stages they proposed three algorithms. Thus, the mechanism has three variations: Classic, DRF and Itinerary. The proposed mechanism is able to process spatial queries without the necessity of periodic beacon transmissions for neighbor table updates or for synchronization. Hence, it can work properly over different types of duty cycle algorithms. However when small duty cycles are used the nodes have less active neighbors. Hence, occurs more often prematurely stopping due to the second halting criterion.

III. PROBLEM IDENTIFICATION AND THE PROPOSED SOLUTION

III.1 Overview

In extension to our previous work [12], we implement the coverage-guaranteed distributed sleep/wake scheduling with adaptive sleep wakeup duty cycle adjustment. Here the cluster formation is formed from the previous work i.e. when an initiator node selects the energy information of the node and selects the cluster heads which has the highest residual energy. Through guaranteed distributed sleep/wake scheduling scheme, we compare entire cluster's energy giving a connection value for each cluster in order to be in connect with the active nodes of other clusters. Once the selected cluster head node receives the joint request J_{REQ} message from member nodes, it sends a joint reply message J_{REP} back to the nodes. Then if the particular cluster head's residual energy is greater than the given threshold value, only one node with higher energy will be in the active mode and the other nodes will be in the sleep mode and also the clusters with lower residual energy will also be goes to sleep mode. Then the CH transmits data to the sink node.

The advantage of the proposed extension is based on the cluster head's residual energy the cluster nodes are decided to be on active mode or in the sleep mode which increases the lifetime of the network.

III.2 Cluster Head Selection

In the cluster head selection initially we select the node with the highest residual energy as the cluster head we do this to prolong the network lifetime. In order to the collect the network's information of the nearest sensor nodes we consider initiator nodes to do this job. By this way we select the cluster heads depending upon the energy information.

Assume that the sink node here has the knowledge about all the sensors nodes and their location in the network. Since we know that the sensor nodes are immobile and also have limited energy. So we determine the cluster head (CH) based upon the residual energy of the nodes which is found out by the initiator nodes (I). In a network, CH uses more energy than other nodes. But the network performance degrades when the energy of the cluster head goes down. To overcome this situation we have to balance the network energy consumption by ensuring that the CH keeps changing in a cluster depending upon its residual energy.

The process of the cluster head selection in the network takes place through the following steps.

- The initiator node I broadcasts a request message for energy (E_{REQ}) with its own residual energy level (RL_{ini}) information to its surrounding nodes.
- The sensor node S_i compares its own energy level (RL_i) with the initiator.
- If $RL_i > RL_{ini}$, then, Si sends a reply message for energy (E_{REP}).

  Else

  S_i waits for cluster head advertisement messages (CH_{ADV}).
- The initiator node (I) selects the cluster head with maximum residual energy and the next initiator node is the node having the second maximum residual energy.
- The initiator node is changed every time when the energy level of the node decreases.
- After CH is selected by the initiator node (I), clusters are formed in the network.
- The nodes in the cluster broadcasts a CH_{ADV} to the CH and CH sends it to the sink node along with the cluster ID.
- A join request message J_{REQ} is transmitted by the member node along with CH_{ADV}.

The transmission range gets minimized since the initiators collect the energy information about the nearest sensors. The nodes having energy greater than the energy level of the initiator ensure minimization of E_{REP} message transmission.

Once the selected cluster head node receives the J_{REQ} message from member nodes, it sends a joint reply message J_{REP} back to the nodes. Then the CH transmits data to the sink node.

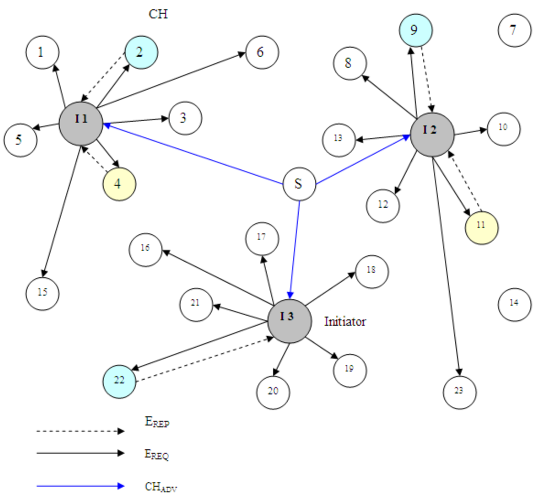

Fig 1: Selection of Cluster Head

In figure 1, the Initiator I1, I2, and I3 sends energy request E_{REQ} to all the surrounding nodes. For example, in this figure, when the node I1 sends E_{REQ} to the nodes, the energy consumption of the nodes is compared with I1. Since node 2 has a higher energy level than I1 and so node 2 is selected as the cluster head and node 4 which has the next highest energy level is selected as the next initiator for cluster head selection.

Similarly, when the node I2 sends E_{REQ} to the nodes, the energy consumption of the nodes is compared with I2. Here, node 9 has the highest energy level and it is selected as the Cluster head and the node 11 which has the next highest energy level is selected as the next initiator for cluster head selection

Then when the node I3 sends E_{REQ} to the surrounding nodes, the energy consumption of the nodes is compared with I3. Since node 22 has the highest energy level, it is selected as the cluster head and no other surrounding node has the next highest energy level other than initiator I3. So, we select node I3 as the next initiator node.

III.3 Sleep/Wake Scheduling

Sleep/Wake scheduling is an every effective process where the network's energy is saved to a maximum extent. In the network once the cluster formation is done then the every cluster starts applying the process of sleep/wake scheduling. In order to save energy, only one or two nodes with highest residual energy in each cluster are required to keep active, while others will be kept on in the sleep mode.

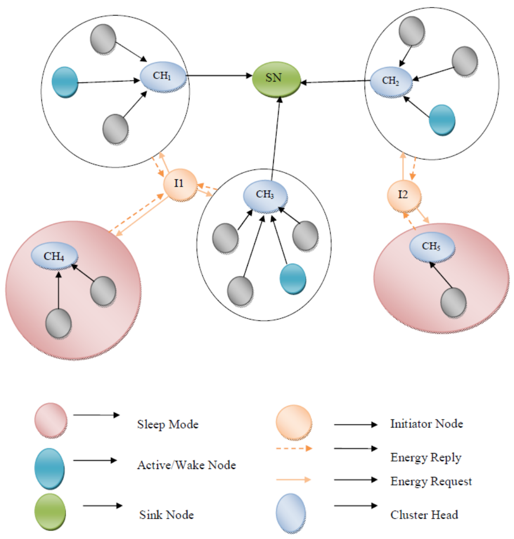

Fig 2: Sleep/Wake Scheduling

At the beginning of this scheduling all the nodes in the cluster will be active in order to analyze the residual energies. This analysis is done to select an active node with the highest residual energy in a cluster. And this active node will undertake the sensing task in a cluster. The node which will undertake the sensing task will be decided by the cluster head. The cluster head sends a WORK message to order the selected node to perform its duty as an active node, moreover, one of which is told to be the head node in the next period. And also the cluster head sends a SLEEP message to all of the rest nodes. When the particular task is completed all the nodes in the sleep mode will send a WORK_REQ to the cluster head to participate in node(s) selection when the next round comes. While receiving WORK_REQs from all the sleeping nodes in the cluster, the head will run the process of selecting active nodes.

In the above fig the initiator node in the network broadcasts an energy request message to all the clusters in the network in order to know the residual energy. Then the entire cluster sends an energy reply to the initiator node. By this way the initiator node collects the residual energy level of the clusters. The initiator node compares the clusters residual energy with the fixed standard threshold value, if any cluster's energy is less than the threshold value than that particular cluster will be sent to the sleep mode. For example consider in the above figure, let the threshold value be 10 and consider the cluster1's energy is level is 12, cluster2's is 13, cluster3's is 11 and let the cluster4's and the cluster5's be 5 and 6. Now in this case after these values are collected by the initiator node it'll compare with the threshold. Because of this reason we can see that only three clusters are transmitting data to the sink node and the 4th and 5th clusters are sent to the sleep mode.

III.3.1 Phases of Sleep/Wake Scheduling

The cycle of sleep/wake scheduling is divided into four phases, namely, sync phase, data phase, sleep phase and report phase. Fig.3 shows the cycle evolution of sleep/wake scheduling. Node H is CH Node and A, B, C and D are NNs.

In sync phase, H broadcasts sync packet, hence each NN can receive sync packet. NN sets its duty cycle and updates its local time based on the information in sync packet. At the remaining time of this cycle, sensor nodes i.e., Node C and D, who do not participate in communication; can go to sleep until next sync phase. So a large amount of energy is conserved. In data phase, node A transmits data packet to B, then B replies ACK to A. After sending the ACK, B sleeps to save energy. If the data period is less than the duration of DATA/ACK, then data period is extended to sleep phase. Finally, the request phase is divided into many time slots. Each NN selects a random slot to send REQ packet to H. The REQ packet has the traffic and routing information about the current node. If it is a winner, H sends a REP (reply) message to validate the REQ packet.

Node A, B, C, D can send the REQ packet to H; H reply REP packet only to the winner node. All the other sensor nodes can go to sleep at the rest time of current cycle. If a NN does not access the channel in the request phase, it will retransmit the REQ packet until the maximal retransmit limits, but the contention window does not increase during the retransmit processes. Due to the dynamic feature of sensor networks, nodes may join or leave a cluster and the cluster is reconstructed to optimize network performance.

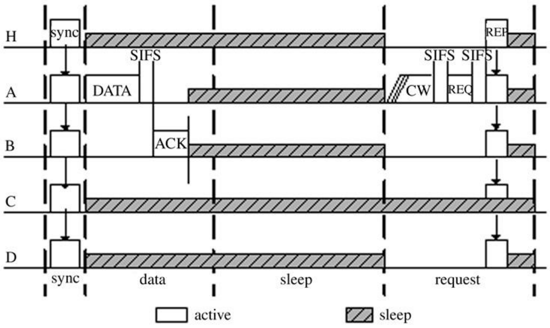

Fig. 3. The cycle evolution of sleep/awake scheduling

III.3.2 Packet structure

In sleep/wake scheduling, there are various types of packets, namely: Sync packet (SYNC), Request packet (REQ) and Reply packet (REP). The format of theses packets are shown in Fig.4. The size of these control packets such as SYNC, REQ and REP, are fixed and small so that energy consumption is less. The CYC LEN field of sync packet has the information of entire cycle length. TS field has the information about how long the sensor node sleeps. CYC LEN and TS field provides the information of duty cycle. All NNs in the range of cluster set their duty cycle based on the information in the sync phase. TM field has the present time of cluster head node. Based on this, all NNs adjust their local time. CN field has the routing information of the nodes involved in communications. If NN has no data to send in next cycle, it will sleep until the sync. The TF field of REQ packet has the traffic information and REQ packet has the CN field, which is similar to the one in SYNC packet. The REP packet has DA field that is based on the address of the winner.

SYNC packet

PT	LEN	CYC LEN	TS	CN	TM	CRC

REQ packet

PT	LEN	DURA	SA	TF	CN	CRC

REP packet

PT	LEN	DURA	DA	CRC

PT: Packet type **LEN:** Packet length **CYC LEN:** Cycle length
TS: Time to sleep **CN:** Communication nodes **SA:** Source address
TF: Traffic load **DURA:** Duration of packet **DA:** Destination address
TM: Current time of CHN

Fig.4. Packet structure of SYNC, REQ and REP

III.3.3 Traffic adaptive duty cycle

Let the length of wake-sleep cycle be L, which is divided into sync, listen, sleep, and report period. Only duration of sleep phase changes and the other phases are constant. The cycle length L needs a proper design to fit for the traffic. If *L* is large, packet information will tend to out of data and the buffer of node tend to overflow. In addition, NNs synchronize with CHN every cycle, the long L may make clock drift too large in a cycle, which leads to communication problems. If L is small, the unnecessary wakeup actions will lead to large energy wasting. Long cycle will lead to long average latency. Latency is limited by applications, so the maximum cycle L is also limited by latency requirement of applications running on the sensor node. Let the duration of sync packet in a cycle be L*sync*, data packet be L*data*, sleep packet be L*sleep*, and request packet be L*request*,

So,

$$L = Lsync + Ldata + Lsleep + Lrequest. \quad (1)$$

For avoiding communication error, we let $L < L_{drift.}$ If the max latency requirement of application is L_{max}, so

$$L < \min\left\{\frac{C-\psi+\mu Tsend}{\lambda} \quad \frac{C-\psi}{\lambda} + Lsend, \frac{1}{BX\theta}, L_{\max}\right\} \quad (2)$$

where C is buffer capacity of current node
λ, μ is average arrival rate
θ is clock drift rate
B is channel bandwidth
L_{drift} is run time for generate 1/*B* clock drift
Ψ is packets in current buffer

So the cycle length is affected by *C, μ,* L_{send} *, B, θ, etc*. In this design, when traffic increases, *Tsleep* decreases, and cycle length L reduces, so sensor nodes have more chance to transmit packets to other nodes to reduce latency and increase throughput. When the traffic reduces, *Tsleep* is increased and sensor nodes spend less time on idle listening for energy saving requirement.

3.4 Algorithm

Initially an initiator node randomly sends an energy request (E_{REQ}) to all the surrounding nodes to collect the residual energy details of the nodes. The initiator node compares its energy with the node's energy. If the node's energy is more than the initiator node then the sensor node (S_i) will send an energy reply (E_{REP}) to the initiator node. Through this way the CH is decided and this CH broadcast the CH_{ADV} to the sensor nodes. Each node sends a joint request (J_{REQ}) to the CH, accepting this request the CH sends a joint reply (J_{REP}) to the sensor nodes and this way the cluster is formed.

In a cluster the CH selects the node with high residual energy and keeps that node in active and sends the remaining nodes into the sleep mode. Then again the initiator node sends an E_{REQ} to the cluster for their residual energy and this energy will be compared with the threshold value. If the energy value is less than the threshold value than the particular cluster will be sent to the sleep mode. And finally the transmission to the sink node will be done by the clusters in the network.

1. In a sensor network consider some nodes as initiator nodes to collect the residual energy details of the sensor nodes.
2. The initiator I1 selects the cluster head based upon the energy level information.
   For each neighbor Ni of I1, i=1,2….r
   If $RL_i > RL_{ini}$, then,
   S_i sends a reply message for energy (E_{REP}).
   Else
   S_i waits for cluster head advertisement messages (CH_{ADV}).
   End For
3. I1 select the node St as CH1 such that RLt = max {RL_i}, i=1,2….r
4. CH1 broadcast a CH_{ADV} to N_i
5. Each node N_i sends request J_{REQ} to CH1.
6. On accepting J_{REP} from the CH, nodes N_i join the cluster.
7. The cluster selects the node with high residual energy and keeps that node in active and sends the remaining nodes into the sleep mode.
8. The initiator node I1 send the energy request to clusters
   If the RL (cluster) > threshold value
   The cluster will be in active mode
   Else
   The cluster will be sent to the sleep mode
9. Transmit the data to the sink node
10. End

IV. SIMULATION RESULTS

The proposed Adaptive Sleep/Wakeup Scheduling Technique (ASST) is evaluated through NS2 [13] simulation. A random network deployed in an area of 500 X 500 m is considered. Initially 100 sensor nodes are placed in square grid area by placing each sensor in a 50x50 grid cell. 10 cluster heads are deployed in the grid region according to our protocol. The sink is assumed to be situated 100 meters away from the above specified area. In the simulation, the channel capacity of mobile hosts is set to the same value: 2 Mbps. The simulated traffic is CBR with UDP source and sink. The number of sources is per cluster is varied fro 1 to 4. Table 1 summarizes the simulation parameters used

Table 1: Simulation Parameters

No. of Nodes	100
Area Size	500 X 500
Mac	802.11
Routing protocol	EEMD
Simulation Time	50 sec
Traffic Source	CBR
Packet Size	512 bytes
Rate	100 to 300kb
Transmission Range	150m
Transmit Power	0.395 w
Receiving power	0.660 w
Idle power	0.035 w
Initial Energy	20.1 Joules
No. of clusters	10

IV.1. Performance Metrics

The performance of ASST technique is compared with SMAC protocol. The performance is evaluated mainly, according to the following metrics.

- **Average end-to-end delay:** The end-to-end-delay is averaged over all surviving data packets from the sources to the destinations.
- **Average Packet Delivery Ratio:** It is the ratio of the number .of packets received successfully and the total number of packets transmitted.
- **Residual Energy:** It is the average residual energy consumed for the data transmission.
- **Throughput:** It is the average rate of successful message delivery over a communication channel.

IV.2. Results

Based on Rate

We vary the transmission sending rate as 100,150,200, 250 and 300 Kb for CBR traffic.

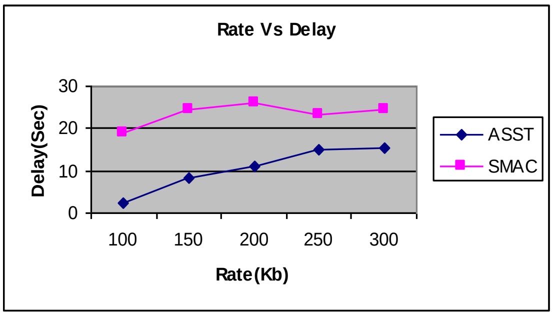

Fig 3: Rate Vs Delay

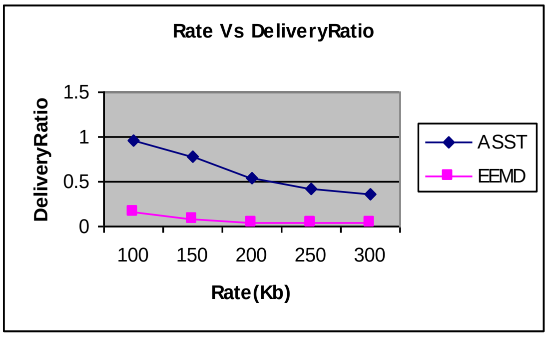

Fig 4: Rate Vs Delivery Ratio

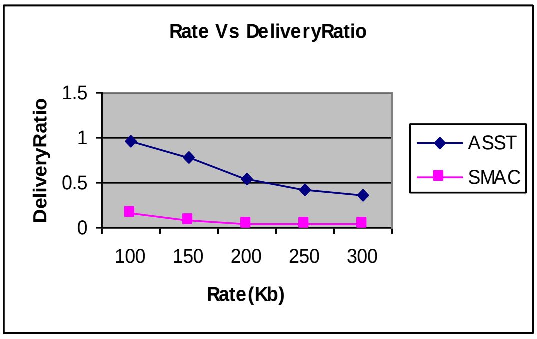

Fig 5: Rate Vs Drop

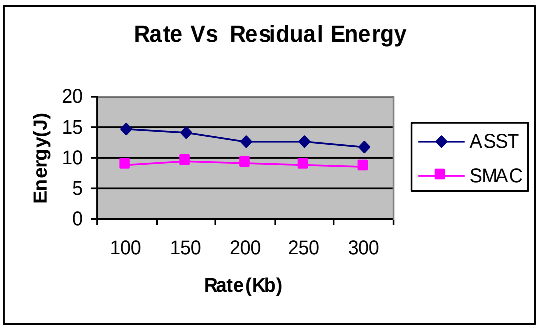

Fig 6: Rate Vs Energy

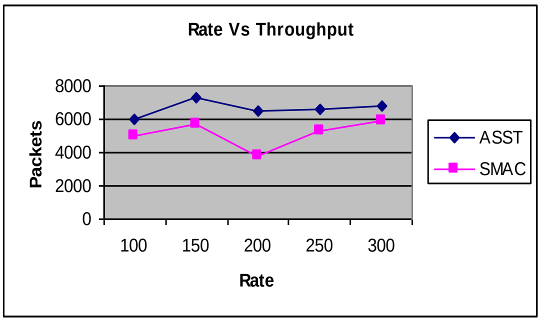

Fig 7: Rate Vs Throughput

From figure 3, we can see that the delay of our proposed ASST is 56.8% less than the existing SMAC technique.

From figure 4, we can see that the delivery ratio of our proposed ASST is 89.8% higher than the existing SMAC technique.

From figure 5, we can see that the packet drop of our proposed ASST is 90.7% less than the existing SMAC technique.

From figure 6, we can see that the energy consumption of our proposed ASST is 31.4% less than the existing SMAC technique.

From figure 7, we can see that the throughput of our proposed ASST is 22.3% higher than the existing SMAC technique.

V. CONCLUSION

In this approach we initially consider some nodes as initiator nodes through these nodes we decide the cluster head. Then these cluster head forms clusters in the network. In these clusters the cluster head selects a node with the highest energy and keep it in active mode and sends the remaining nodes into the sleep mode. Then again the initiator nodes collect the residual energies details of these clusters and compare with the standard threshold value. If the cluster energy is less than the threshold value then that particular cluster will be sent into the sleep mode and a traffic adaptive sleep-wakeup scheduling technique is provided. At the last the transmission will be done by the active clusters to the sink node. Through this way it is possible to decrease the energy consumption of the network and increase the network's life. By simulation results, we have shown that the proposed scheduling technique increases the average residual energy and throughput

REFERENCES

[1] Guofang Nan, Guanxiong Shi, Zhifei Mao and Minqiang Li, "CDSWS: coverage-guaranteed distributed sleep/ wake scheduling for wireless sensor networks", EURASIP Journal on Wireless Communications and Networking 2012.

[2] Mihaela Cardei and Ding-Zhu Du,"ImprovingWireless Sensor Network Lifetime through Power Aware Organization", ©2005 Springer Science + Business Media, Inc.

[3] Ahmed Ben Nacef, Sidi-Mohamed Senouci, Yacine Ghamri-Doudane and Andre-Luc Beylot, "A Cooperative Low Power Mac Protocol for Wireless Sensor Networks", Communications (ICC), 2011 IEEE International Conference on. IEEE, 2011 APA.

[4] Soumya Ray, Subhasis Dash, Nachiketa Tarasia, Anuja Ajay and Amulya Ratan Swain, "Congestion-less Energy Aware Token Based MAC Protocol Integrated with Sleep Scheduling for Wireless Sensor Networks", Proceedings of the World Congress on Engineering 2011 Vol II.

[5] Giuseppe Anastasi, Marco Conti and Mario Di Francesco, "Extending the Lifetime of Wireless Sensor Networks through Adaptive Sleep", IEEE Transactions on Industrustrial Informatics, 2009.

[6] Sabitha Ramakrishnan and Thangavelu Thyagarajan, "Performance Enhancement of Fuzzy Logic Based Transmission Power Control in Wireless Sensor Networks using Markov Based RSSI Prediction", European Journal of Scientific Research ISSN Vol.59 No.1 (2011).

[7] Binu G S and Dr. K Poulose Jacob, "Channel Adaptive MAC Protocol with Traffic-Aware Distributed Power Management in Wireless Sensor Networks-Some Performance Issues", IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.7, July 2010.

[8] Rodolfo de Paz and Dirk Pesch, "DCLA: A Duty-Cycle Learning Algorithm for IEEE 802.15.4 Beacon-Enabled WSNs", Ad Hoc Networks Springer Berlin Heidelberg, 2010.

[9] LIU Hao, YAO Guoliang and WU Jianhui, "CBA-MAC: An Adaptive MAC Protocol over Clustering Based Wireless Sensor Networks", Chinese Journal of Electronics Vol.20, No.1, Jan. 2011.

[10] Shouwen Lai, "Duty-CycledWireless Sensor Networks: Wakeup Scheduling, Routing, and Broadcasting", Virginia University 2010.

[11] Rone Ilídio da Silva, Daniel Fernandes Macedo and José Marcos S. Nogueira, "Duty cycle aware spatial query processing in Wireless Sensor Networks", © 2012 Elsevier B.V.

[12] Sasikala.V and Chandrasekar.C, "Cluster Based Sleep Wakeup Scheduling Technique for WSN" , International Journal of Computer Applications, May, 2013.

[13] Network Simulator: http://www.isi.edu/nsnam/ns