2009 First International Conference on Networks & Communications

ENERGY EFFICIENT WIRELESS SENSOR MAC PROTOCOL FOR COLLISION AVOIDANCE

Abhishek Samanta*, Dripto Bakshi†, Amitava Mukherjee‡, Mita Nasipuri§
*Jadavpur University,Kolkata
Email:avisheksamantajunior@gmail.com
†Jadavpur University,Kolkata
Email: bakshi.dripto@gmail.com
‡IBM India Pvt Ltd.,Kolkata
Email: amitava.mukherjee@in.ibm.com
§Jadavpur University,Kolkata
Email: mnasipuri@cse.jdvu.ac.in

***Abstract*—In sensor networks communication by broadcast method involves many hazards, especially collision. Several MAC layer protocols have been proposed to resolve the problem of collision namely ARBP, where the best achieved success rate is 90%. We hereby propose a MAC protocol which achieves a greater success rate (Success rate is defined as the percentage of delivered packets at the source reaching the destination successfully) by reducing the number of collisions, but by trading off the average propagation delay of transmission. Our proposed protocols are also shown to be more energy efficient in terms of energy dissipation per message delivery, compared to the currently existing protocol.**

***Keywords*-Wireless sensor networks, Energy efficient, Data propagation, Single path, Multi path, Collision Avoidance, Success rate**

I. INTRODUCTION

Sensor networks are particularly useful in collecting data from inaccessible terrains which serve various purposes in further investigation. Sensor network comprises of a large number of nodes (often termed as motes) which are randomly distributed very densely in the area concerned. The data collected by each node is transmitted to subsequent nodes and thus finally resulting in the reporting of the data to the sink which can be considered as a destination for delivering data.

A. Motivation

The common methods employed to resolve the MAC layer problems cannot be employed in sensor networks. CSMA cannot be employed [1] because the csma-based protocols involve broadcasting of control messages between the sender and the receiver, in order to enable the sender to acquire the transmission media, and these eventual broadcast of messages result in a collision between these control messages.

Existing backoff schemes:

In order to resolve the Mac-layer problem of collision a few backoff schemes have been designed, namely SRBP (Simple Random Backoff Protocol), ARBP (Adaptive Random Backoff Protocol) and RARBP (Range Adaptive Random Backoff Protocol) [2]. In the backoff protocols the broadcasts are delayed by a certain backoff period,i.e. the nodes assume a random backoff time and then transmit the data packet it wants to transmit. Thus simultaneous broadcasts and consequent interference of data resulting in collision is prevented, since the broadcasts of the several nodes are spread over time.

The backoff schemes when applied on the AODV protocol significantly improve the success rate (which is defined as the percentage of broadcasted packets reaching the final destination). Amongst all the backoff schemes used till date ARBP achieves the highest success rate (i.e. success rate of 90%). We hereby propose a backoff scheme which achieves a success rate of around 95%, which is significantly higher than those achieved by the existing protocols. The backoff schemes are adopted to prevent collisions by spreading the broadcasts over time. Since the nodes adopt different backoff periods before transmitting a data packet, simultaneous broadcast is prevented and hence reducing the number of collisions. The continuous set from which the backeoffs are selected (T_{min},T_{max}) varies from one protocol to another[2].

But these protocols ignore the fact that more than one nodes may select the same backoff period, or any node can choose a backoff period in such a way that it starts to transmit its packet of data before another previously transmitted data packet has finished transmission, thereby resulting in collisions in both cases. This problem has been approached from two directions. In one of our schemes, it has been ensured that nodes always choose a backoff period in such a way so as to overcome the above mentioned deficiencies. In other scheme, the backflow of packets (the flow of packets in the direction opposite to the direction of destination node,ie sink) has been controlled. Both of these schemes ensure the low incidence of collisions and thereby increase the success rate of overall protocol.

978-0-7695-3924-9/09 $26.00 © 2009 IEEE
DOI 10.1109/NetCoM.2009.28

II. PROPOSED MAC PROTOCOLS

A. Informed Backoff selection protocol (IBSP)

The protocol primarily consists of two phases. During the first phase the backoff period selected by a node for data transmission is informed to the neighboring nodes in the radius of transmission of that node, and thus the other nodes select their respective backoff-periods accordingly. Thus this internodal sharing of information about the backoff periods chosen during data packet transmission, further eliminates the possibility of coincident transmissions, and thereby reduced collisions. During the second phase actual data transmission takes place.

1) First phase: In the first phase when a node is ready to broadcast a data packet, it first sends a control packet consisting of a MAC header in which the backoff period for data transmission is appended. The backoff scheme used for the the transmission of control packets is governed by ARBP, where the backoff period is selected from the continuous set (T_{min},T_{max}). Here T_{min} is the minimum time taken for the node to node transfer of packets and T_{max} is calculated by the formula proposed in the ARBP scheme [2].

Selection of backoff for data transmission takes place according to the following scheme. Whenever a node transmits the control packet to its neighboring nodes, the nodes on receiving the control packet, reads the backoff selected by the sender(say T), and store it in their respective memories. Then each of the receiving nodes mark the zone ($T - T_{min}$, $T + T_{min}$) as the forbidden zone,i.e. no backoff period further selected will lie in this zone. In this is way the next node (as per the ARBP scheme) selects a backoff for data transmission and informs it to the other neighboring nodes, via the transmission of control packets. In this way, by sharing of control packets, the subsequent sharing of information about the backoff time selected by the nodes takes place. The backoff periods for data transmission are selected from the continuous set (T_{min}+T_{max}, $2T_{max}$). This set is selected because of the following reason. During the control packet transmission that follows ARBP scheme, as mentioned earlier, the maximum backoff that can be selected is equal to T_{max}. Now since the maximum time for node to node transmission is T_{min}, so the transmission of control packets must be over by the time ($T_{max} + T_{min}$). Again since the data packet transmission also follows the ARBP scheme, the backoff period selection must take place from a continuous set having a width of ($T_{max} - T_{min}$). Thus the backoff period selection for data transmission takes place from the set (T_{max}+T_{min}, T_{max}+T_{min}+T_{max}−T_{min}),i.e. ($T_{max} + T_{min}$, $2T_{max}$).

The node receiving the data packet first is the first node to let its neighboring nodes know about the backoff, that it is going to take during data propagation in second phase, selected by it. Now when the first node as per the ARBP scheme informs all the other nodes in its neighborhood (i.e. range of radio transmission) about the backoff period it has selected for phase II,i.e. the data packet transmission phase,(say T), then all the informed nodes mark ($T + T_{min}$, $T - T_{min}$) as the forbidden zone while choosing their backoffs, because if any node selects a back off in the above mentioned zone then a collision is bound to occur with the packet delivered by the node which has chosen a backoff of T.

Now again the second node, as per the ARBP scheme, chooses a back off period from the set ($T_{max} + T_{min}$, $2T_{max}$) apart from the forbidden zone created by the first. The backoff thus selected by the second node is informed to all the nodes in its range of radio transmission. When the third node, as per the ARBP scheme, chooses its backoff for data packet transmission, it chooses the backoff from

Algorithm 1: IBSP

Tmin → minimum time taken for node to node delivery of packets
Tmax → maximum backoff time optimized in order to reduce the number of collisions and is calculated according to ARBP scheme
eligible_set → a set from which to select a backoff from for the data transmission phase (second phase)
queue_packet → queue of packets need to be broadcasted out

PHASE I

eligible_set ⇐ [$T_{max} + T_{min}$, $2T_{max}$]
backoff_phase_1 ⇐ k ϵ [T_{min},T_{max}] according to ARBP
backoff_phase_2 ⇐ t ϵ eligible_set according to ARBP
wait for backoff_phase_1
if no packet from neighbor received **then**
    Send backoff_phase_2 to all neighbors
else
    if the packet is a data packet **then**
        enque(neighbor_packet to queue_packet)
    else
        t_n ⇐ backoff selected by neighboring node for phaseII
    end if
    eligible_set⇐eligible_set−[t_n-T_{min}, t_n+T_{min}]
    goto step 2
end if

PHASE II

wait for backoff_phase_2
packet_data ⇐ deque(queue_packet)
broadcast(packet_data)
if data packet from neighbor received **then**
    neighbor_packet_hopcount⇐hop_count_level_of_node
    enque(neighbor_packet to queue_packet)
end if

the set ($T_{max} + T_{min}$, $2T_{max}$) apart from the forbidden zones specified by the first two nodes. In this way, any node chooses its back off for data packet transmission phase from the set ($T_{max}+T_{min}$,$2T_{max}$) apart from the forbidden zones created by the nodes preceding it, as per the ARBP scheme.

Thus for nodes in the range of transmission of each other, there is the sharing of information regarding the back off periods selected for data transmission of phase II and thus collision is minimum.

We have mentioned earlier that the information sharing about chosen backoff and hence careful subsequent back off selection amongst the nodes within each others range of radio transmission. Let us say that these nodes form a group. Now any node may be part of different groups. Now all nodes in a particular group are well aware of the back offs chosen by the other nodes of that group and accordingly have selected their own backoffs so as to avoid collision. Now let us consider that a set of nodes N_i be part of two groups G_1 and G_2. Now there is sharing of information regarding backoff selection in group G_1 as well as in group G_2. But since these two groups share a few nodes between them, so there will also be an intra-group sharing of information. Here the common nodes serve as the link between the two individually coordinated groups, and this leads to an unified coordination amongst all nodes of the groups G_1 and G_2. Since nodes are more or less always shared by two or more groups so the intra-group sharing of information regarding the chosen backoffs and hence careful coordination in backoff selection eventually leads to an inter-group coordination as well. In other words coordination is achieved amongst all nodes trying to send data packets at any instant of time,i.e. the backoffs are so selected and coordinated that the number of collisions is minimum.

2) Second phase: In this phase the actual data transmission takes place. The nodes assume the backoffs previously chosen by them in phase I, and eventually deliver the data packets, thus reducing the number of collision to the minimum.

B. Direction Aware IBSP (DAIBSP)

When the topology is ready to be used the sink node broadcasts a beacon packet consisting only a field which keeps record of hop count and this beacon packet is broadcasted by each node to their neighbors. When a node receives the beacon packet, it compares its present hop count level (which is initially set to -1, which indicates that the node has not received a packet yet) with the hop count field in the beacon packet; if the hop count field value in the beacon packet is greater than or equal to present hop count level of the node then this information is discarded. Else the hop count level of the receiving node is overwritten by the hop count value of the beacon packet and the beacon packet is rebroadcasted, after setting the hop count level in the packet to the hop count level of the node. Initially when transmitted by sink hop information in beacon node is set to zero and in each hop the field is incremented by one. This beacon packet is broadcasted according to the IBSP protocol described above.

Every node maintains a table consisting of two fields,viz hop count and a timer. At any moment the hop count of a node is associated with the minimum entry from the hop count table with live timer, is not overwritten until there is any entry in the table having live timer. While transmitting

Algorithm 2: DAIBSP

Tmin → minimum time taken for node to node delivery
Tmax → maximum backoff time optimized by ARBP
eligible_set → a set from which to select a backoff from for the data transmission phase (second phase)
queue_packet → queue of packets need to be broadcasted out

Building phase

```
recieve(sink_packet)
if hop_count_of_node ≤ sink_packet_hopcount then
    hop_count_level_of_node = sink_packet_hopcount
    sink_packet_hopcount ⇐ sink_packet_hopcount + 1
    broadcast(sink_packet_hopcount)
end if
loop
    call phase_I_DAIBSP // PHASE I
    call phase_II_DAIBSP // PHASE II
end loop
```

Algorithm 3: subroutine phase_I_DAIBSP

```
eligible_set ⇐ [T_max + T_min, 2T_max]
backoff_phase_1 ⇐ k ϵ [T_min,T_max] by ARBP scheme
backoff_phase_2 ⇐ t ϵ eligible_set according to ARBP
wait for backoff_phase_1
if no packet from neighbor received then
    Send backoff_phase_2 to all neighbors
else
    call handler_packets_from_neighbor
    if the packet is a data packet then
        enque(neighbor_packet to queue_packet)
    else
        t_n ⇐backoff selected by neighboring node for phaseII
    end if
    eligible_set⇐eligible_set−[t_n-T_min, t_n+T_min]
    goto step 2
end if
return
```

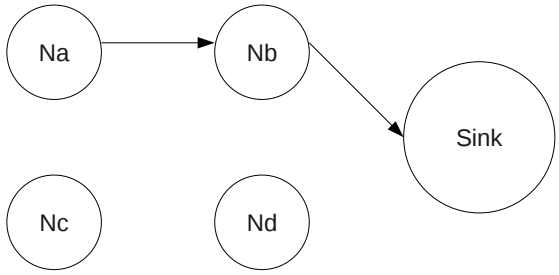

Figure 1: Path of data propagation before failure of N_b

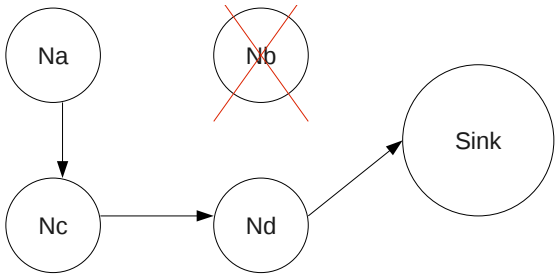

Figure 2: Path of data propagation after failuer of N_b

a packet the hop count of the node is included in the packet. a node does not transmit any packet until it knows its hop count level. Thus the hop count level of each node is dynamically determined during the data transmission phase after it has been initialized during the beacon transmission cycle carried out earlier, by the sink. When a node receives a packet with hop count level less than its present hop count level then the hop count field of the packet is noted down in the table and the packet is not broadcasted any more and hence discarded. The packet transmission phase is done according to the IBSP protocol. In this way the packets always are forwarded from the source to the sink and hence undesirable backflow (i.e. the flow of data in the opposite to the source-sink direction) of information is hindred. Thus the extra consumption of energy that results from the subsequent collisions due to the above mentioned backflow is avoided.

The hop count level table is maintained to protect the network from debacle when some unforeseen catastrophe happens. Let us consider 3 nodes, N_a, N_b, N_c. N_a is at a distance d_b from the sink through N_b and at a distance d_c through the node N_c, where $d_b < d_c$. So the hop count of N_a is d_b [Figure 1]. Now if N_b dies, the packets emanated from N_a is dropped by N_c since the hop count level in the packets is less than that of N_c. This happens only till the timer associated with d_b is alive. After the timer associated with d_b expires, N_a assigns d_c as its hop count and then the data transaction is again restored [Figure 2].

Algorithm 4: subroutine phase_II_DAIBSP

```
wait for backoff_phase_2
packet_data ⇐ deque(queue_packet)
broadcast(packet_data)
if packet from neighbor received then
    call handler_packets_from_neighbor
    if the packet is a data packet then
        enque(neighbor_packet to queue_packet)
    end if
end if
return
```

III. Experimental Evaluation

To evaluate the performance of the proposed protocol we have done simulation analysis in NS−2 (network simulator). We start our experimentation by dropping nϵ[500,1500] within a smart dust plane of 500m×500m in dimension. We also generated 2000 events by randomly selecting a particle in the network for each event. With event generation rate, $\lambda \epsilon \{5,8,10\}$. For DAIBSP a table with 10 least hop counts were maintained with a initial timer value of 2sec. The transmission range for one node has been set to 50 m. We here defined the success rate as

$$\text{Success rate} = \frac{data\ received\ by\ sink}{data\ sent\ by\ source\ node} \times 100\%$$

From our first experiment we have found that the success rate for a sensor network increases about 4% when we use IBSP instead of ARBP and DAIBSP increases success rate by 1% more on IBSP protocol. The success rate increases drastically as the node density goes from 500 to 1000 and decreases by little bit as the node density increases further. With node density 1000, the success rate reaches around 95%. The result remains almost same when simulated with $\lambda = 8$. The average delay, on the other hand, increases by 80% over normal ARBP.
With lower node density the success rate of IBSP protocol was almost equal to that of ARBP. This can be explained by the fact that, at lower node density the expectation of two nodes being in the range of each other, is less than that incase of higher concentration of nodes. So at n = 500

Algorithm 5: handler_packets_from_neighbor

```
pkt → packet from neighbor
enlist(pkt,t) → put hop_count_in_pkt+1 associating it
to a timer with value t in the hop count table
enlist(pkt,T)
if hop_count_of_node < hop_count_in_pkt then
    hop_count_of_node ⇐ hop_count_in_pkt + 1
end if
```

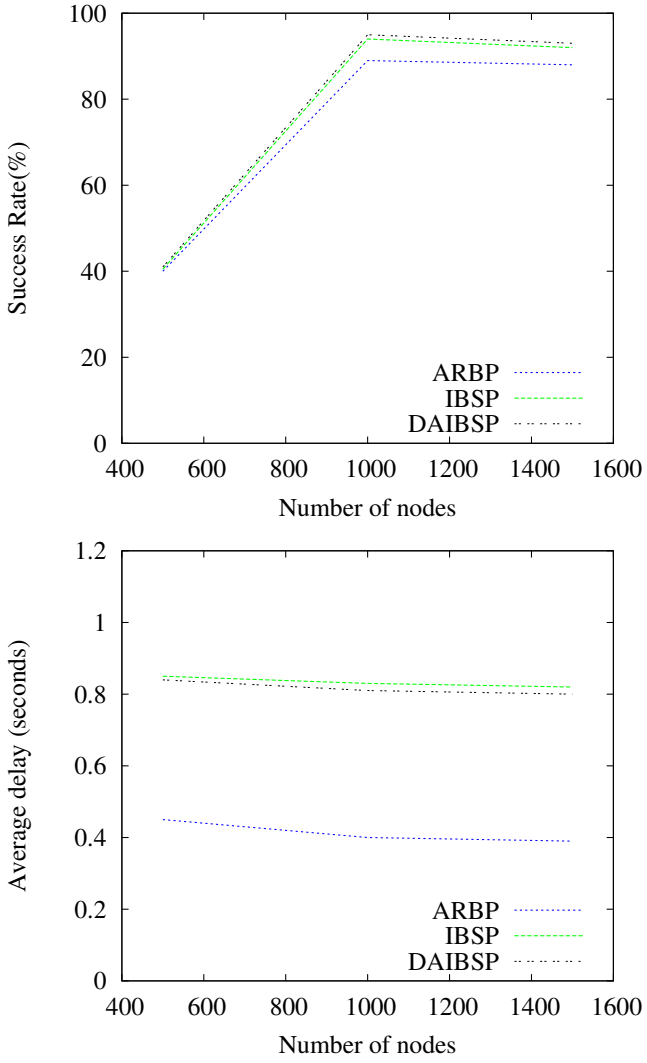

Figure 3: Success rate and average delay for varing node densities (n ϵ [500,1500]) with $\lambda = 5$

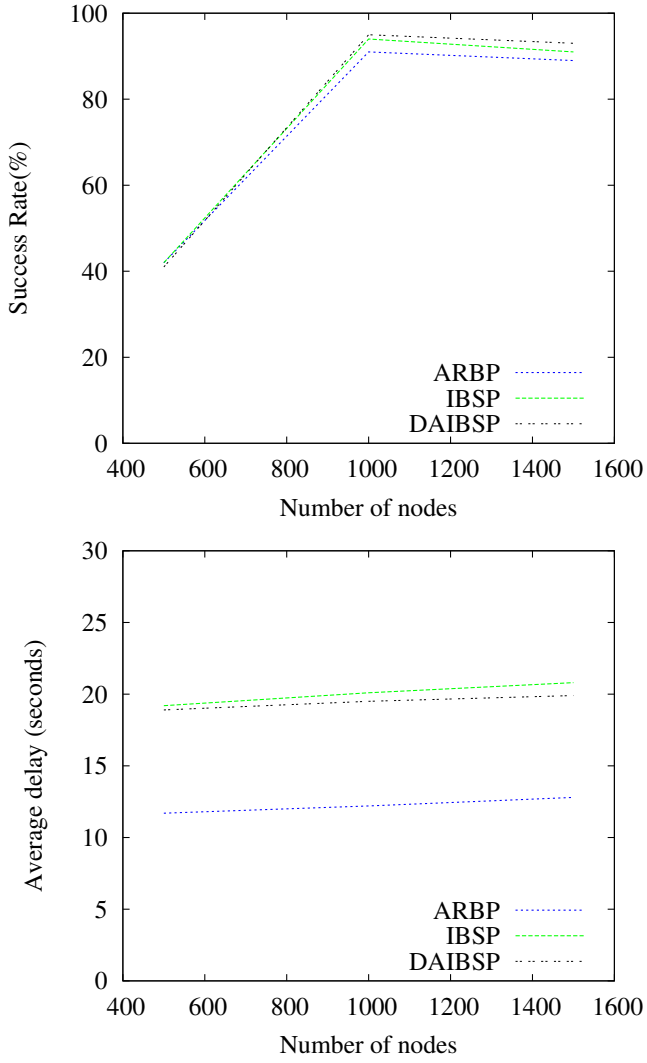

Figure 4: Success rate and average delay for varing node densities (n ϵ [500,1500]) with $\lambda = 8$

the lower amount of success rate is mainly because of the unreachability of nodes rather than the collision of packets. But as the concentration of the nodes gets increased the reachability problem is overtaken by the problem generated by the overcrowded traffic which causes packets to drop.

This is where IBSP performs better than ARBP. But as the n increases to 1500, the message traffic becomes so high that the beacon packets from the nodes startes colliding more in the first phase of information sharing, due to which the the neighbors of a node does not get properly informed about the backoff taken by the node. Due to this miscommunication the success rate of IBSP decreases to around 90%.

But both in ARBP as well as in IBSP the data packet can go anywhere in the network irrespective of the direction of the sink, as long as the packet is not moving in a circular path. This adhoc movements of data packets increase probability of two packets getting collided. To avoid this, DAIBSP only allows movement of packets in the direction of sink only. This scheme ensures that a packet is confined within the area having radius equal to the hop count of the node where the packet have been emanated from. Due to this reason the success rate of DAIBSP is even greater than IBSP and since it does not take any extra transmission on simple IBSP to share information, the average delay remained almost same. Simple IBSP scheme observes larger packet drops [Figure 5] because phase I of the scheme when no effective information sharing takes place, follows the ARBP scheme, which inherently has 90% success rate. In phase II, when actual data transmission takes place, some of the data packets still managed to have collision. So for every data packet transmission the number of colliding packets gets increased. But since DAIBSP inhibits backflow of information, the

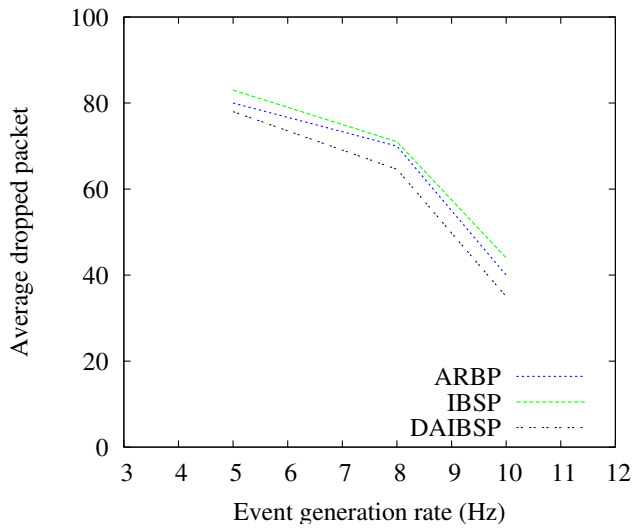

Figure 5: Average number of dropped packets over varing event generation rate with n = 1000

number of colliding packets drastically falls down.

IV. Conclusion

The proposed MAC protocol further resolves the problem of collision that is inherent to broadcasting in sensor networks. In our protocol we tried to incorporate some techniques which were previously used in higher levels in network stack, in MAC layer. Our protocol is first of its kind to make MAC layer handle the hop count information and there by incorporates source-sink direction sensing ability in broadcasting protocols. The protocol reduces collision to a great degree (95% success rate approx) which is a clear improvement over the existing collision resolving protocols (90% success rate or reliability − the best achieved reliability till date). The reduced number of collisions leads to greater energy efficiency by reducing the number of dropped packets, with trade off with the average delay of transmissions.

Reducing the time delay trade off calls for future deliberation.

References

[1] Iiker Demirkol, Cem Ersoy, Fatih Alagoz, *MAC protocols for wireless sensor networks: a survey* IEEE Communications Magazine, vol. 44, iss. 4, pp. 115-121, 2006

[2] Ioannis Chatzigiannakis , Athanassios Kinalis , Sotiris Nikoletseus,*Wireless Sensor Network Protocols for efficient collision avoidance in Multi path Data Propagation* in Proceedings of the 1st ACM international workshop on Performance evaluation of wireless ad hoc, sensor, and ubiquitous networks, 2004

[3] A. Boukerche, R. Pazzi and R. Araujo,*A nove fault tolerant and energy-aware based algorithm for wireless sensor network* in Algorithmic Aspects of Wireless Sensor Networks , 2004.

[4] Chritos H. Papadimitriou* and David Ratajczak,*On a Conjecture Related to Geometric Routing* in algorithmic aspect of wireless sensor network 2004

[5] AN-SWOL HU and SERGIO D. SERVETTO,*Algorithm aspects of the time synchronization problem in large-scale sensor network* in Mobile Networks and Applications, 2005

[6] Tijs van Dam, Koen Langendoen, *An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks* in The First ACM Conference on Embedded Networked Sensor Systems (SenSys 2003),

[7] Koen Langendoen, Niels Reijers, *Distributed localization in wireless sensor networks: a quantitative comparison* in Computer Networks (Elsevier), special issue on Wireless Sensor Networks, November 2003

[8] Seungkyu Bac, Dongho Kwak, Cheeha Kim, *Traffic-Aware MAC Protocol Using Adaptive Duty Cycle for Wireless Sensor Networks* in The International Conference on Information Networking, 2007

[9] Sangik Cha, Jaehoon Ko, Soonmok Kwon and Cheeha Kim, *3-hop Ahead Path Reservation Scheme for Expedite Traffic in Wireless Sensor Networks* in International Conference on Information Networking (ICOIN), 2009

[10] The network simulator – ns2. http://www.isi.edu/nsnam/ns/.