Investigation of Collision Avoidance and Localization in WSNs

Iman Morsi, Mohamed Essam Khedr, Alaa Hassan Abd El-Atif

Department of Communication and Electronics Engineering, Faculty of Engineering and Technology – Arab Academy for Science, Technology & Maritime Transport, Alexandria, Egypt

drimanmorsi@yahoo.com, khedr@aast.edu, alaa.hassan.1987@gmail.com

Abstract

The most important two problems in Wireless Sensor Network (WSN) is localization and collision avoidance of the transferred packets. A solution for these obstacles is introduced in this work. Localization is solved by upgrading anchor position accuracy using a global positioning system (GPS) and position estimates. Collision avoidance obstacle is discussed in the simulation of WSN and studying the effect of collision occurrence during communication between wireless sensor nodes.

Keywords

Wireless Sensor Networks (WSN), Localization, Collision avoidance, Global Positioning System (GPS), Differential Global Positioning System (DGPS)

1. Introduction

Wireless communication is used widely all over the world. The demands of the society increase the number of applications of wireless communication. It has important applications in all aspects such as: industrial, medical, electrical….etc. Wireless communication allows humans to use cellular phone, communication radio, wireless networks, and short range communication….etc. One of the exciting applications of wireless communication is Wireless Sensor Networks (WSNs), which are distributed low power nodes with sensors to collect the specific information and use this information to take specific action. Localization is essential whatever uniform or random. Localization algorithms and schemes try to determine the location of the sensor node with certain error. Sometimes GPS is used to determine the location of some nodes in outdoor systems. These nodes that contain GPS receivers are called anchor nodes whereas the other nodes are ordinary nodes without GPS receivers in order to decrease the economic cost of the GPS receivers [1, 2, 3]. Differential GPS (DGPS) is advanced GPS which can achieve few decimeters accuracy. The type of GPS receiver is decided according to the application, for example, in environmental issues if the model used to collect temperature or pressure or humidity using sensors on the nodes, high accuracy is not needed because the temperature or humidity or pressure is measured over wide range area [4]. High accuracy is needed in military applications where DGPS is used [5].

Collision is what happens when two nodes transmit packets at the same time, packets may be destroyed [6]. An access method called Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA), in which each node listens to the media before transmitting data to avoid collision, is used [7]. The aim of this paper is to achieve high accuracy location with less energy and avoid collision of packets in the communication between WSN nodes [8, 9].

DOI : 10.5121/ijcnc.2012.4604

The paper is organized as follows: Section 2 shows the related work. Section 3 explains the proposed system model and shows the mathematical equations used to increase the accuracy of the localization in the presence of angle estimates and distance estimates. Section 4 describes collision avoidance scheme. Section 5 shows different network topologies produced by simulation using discrete event simulator. Section 6 discusses the simulation results showing different network topologies, and finally section 7 paper conclusion and future work.

2. Related Work

Localization schemes are divided into range based or range free. The localization scheme is range based where distance is determined, and then location is calculated using geometric principles [10]. As was mentioned previously the anchor nodes know their locations using GPS receivers which depends on the triangulation principle in order to detect the locations. It is important to increase the localization accuracy and it differs from system to another, indoor or outdoor, line-of-sight (LOS) or non-line -of- sight (NLOS) [11]. NLOS is radio transmission between a transmitter and a receiver, where an obstacle exists in between them as shown in figure 1. This obstacle can be something by nature for example; hill, tree, mountain, buildings, concrete obstacle, and some of these obstacles can absorb the signal or reflect some frequencies. To investigate localization, LOS condition is assumed, the NLOS error is defined as extra distance between the receiver and the transmitter.

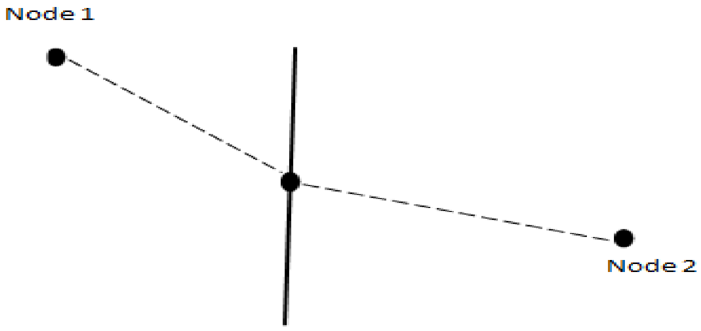


Figure 1. Two nodes in NLOS condition.

3. Proposed System Model

The model consists of N number of nodes. The percentages of these nodes are anchors (with GPS receivers) and others are ordinary nodes (without GPS receivers). Anchors can communicate through radio range to send information collected by each anchor and to send their locations to each other so the ordinary nodes can calculate estimated locations. In figure 2 four anchor nodes, anchor i, anchor j, anchor k and anchor l are presented. Each anchor node has global position coordinates for example node i coordinates are (xi , yi, zi) .Each anchor can measure angle of arrival (azimuth and elevation) [12]. For example $\varphi i,j$ azimuth angle of the signal which is transmitted from anchor j and received at anchor i, elevation angle $\alpha\ i,j$ of the signal transmitted from anchor j and received at anchor i. The distance between any two anchor nodes is called Euclidean distance and can be calculated using equation (1):

$$di,j = \sqrt{(xi - xj)^2 + (yi - yj)^2 + (zi - zj)^2} \qquad (1)$$

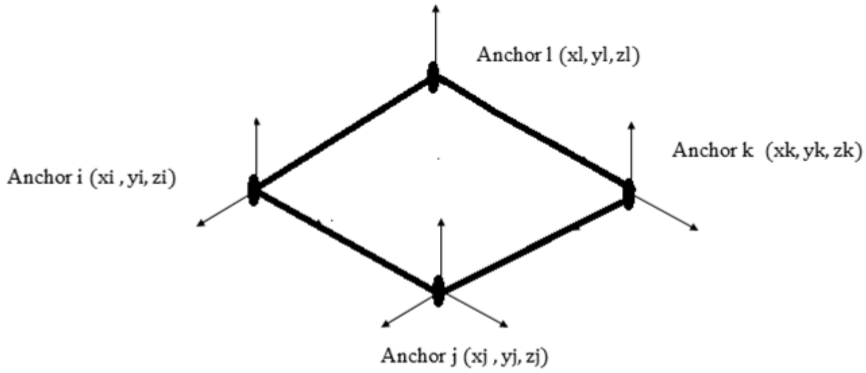


Figure 2. Four anchor nodes locations in LOS condition.

In order to calculate the elevation angle $\alpha i,j$

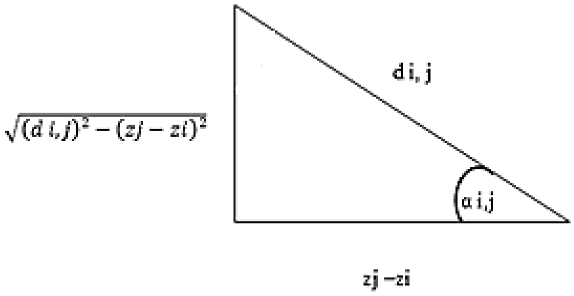


Figure 3. Elevation angle.

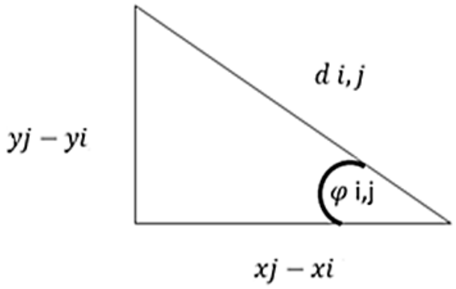


Figure 4. Azimuth angle.

From figure 3 cosine the elevation angle = adjacent divided by hypotenuse producing equation (2). By taking the inverse the elevation angle value can be calculated as given in equation (3)

$$\cos\alpha i,j = \frac{zj-zi}{dij} \tag{2}$$

$$\alpha\ i,j = \cos^{-1}\ \frac{zj-zi}{dij} \tag{3}$$

From figure 4 the calculation of di,j & $\varphi i,j$ equals to equation (4) & (5)

$$di,j = \sqrt{(xj-xi)^2+(yj-yi)^2} \tag{4}$$

cos $\varphi i,j$ = adjacent divided by hypotenuse

$$\cos\ \varphi i,j \quad = (xj-xi)/(d\ i,j\ ) \tag{5}$$

Equation (6) can be deduced by substituting (4) in (5)

$$cos\ \ \varphi i,j = \frac{\mathrm{xj-xi}}{\sqrt{(xj-xi)+(yj-yi)^2}} \tag{6}$$

Therefore

$$xj-xi\ =\ \sqrt{(xj-xi)^2+(yj-yi)^2}\ \times\ cos\ \ (\varphi i,j) \tag{7}$$

From figure 3:

$$\sin\ \alpha\,\mathrm{i,j} = \frac{\sqrt{\ ((d\ i,j)^2-(zj-zi)^2)\ \ )}}{di,j} \tag{8}$$

Therefore

$$\mathrm{di,j} \times \sin\ \alpha\ \mathrm{i,j} = \sqrt{(\mathrm{d\ i,j})^2 \text{-} (\mathrm{zj\text{-}zi})^2} \tag{9}$$

Therefore

$$di,j \times \sin\ \alpha\ \mathrm{i,j} = \sqrt{(xj - xi)^2 + (yj - yi)^2} \tag{10}$$

equation (7) can be written as

$$xj - xi = di,j \times \sin\ \alpha\ \mathrm{i,j}\ \times\ cos\ \ \varphi i,j \tag{11}$$

If the estimation error is taken into consideration

$\mathrm{n}_{\hat{\mathrm{d}}\ \mathrm{ij}}$ is the estimation error for distance d i, j
$\mathrm{n}_{\widehat{\varphi}\ \mathrm{ij}}$ is the estimation error for azimuth angle φi, j
$\mathrm{n}_{\hat{\alpha}\ \mathrm{ij}}$ is the estimation error for elevation angle α i, j

The distance estimate $\hat{d}_{\ ij}$ can be written as distance d i, j added to the estimation error $n_{\hat{d}\ ij}$ as shown in equation (12) .The azimuth estimate $\widehat{\varphi}_{\ ij}$ can be written as an azimuth angle $\varphi i,j$ added to the estimation error $n_{\widehat{\varphi}\ ij}$ as shown in equation (13) .The elevation estimate $\hat{\alpha}_{\ ij}$ can be written as elevation angle $\alpha\ i,j$ added to the estimation error $n_{\hat{\alpha}\ ij}$ as shown in equation (14) [1].

$$\hat{d}_{\ ij} =\ d\ i,j + n_{\hat{d}\ ij} \tag{12}$$

$$\widehat{\varphi}_{\ ij} = \varphi i,j + n_{\widehat{\varphi}\ ij} \tag{13}$$

$$\hat{\alpha}_{\ ij} = \alpha\ i,j + n_{\hat{\alpha}\ ij} \tag{14}$$

From (12)

$$d\ i,j = \hat{d}_{\ ij} \text{ - } n_{\hat{d}\ ij} \tag{15}$$

From (13)

$$\varphi i,j\ =\ \widehat{\varphi}_{\ ij} \text{ - } n_{\widehat{\varphi}\ ij} \tag{16}$$

From (14)

$$\alpha\ i,j =\ \hat{\alpha}_{\ ij} \text{ - } n_{\hat{\alpha}\ ij} \tag{17}$$

Equation (18) is extracted by substituting equations (15) (16) (17) in equation (11)

$$xj - xi = \left(\hat{d}_{\ ij} -\ n_{\hat{d}\ ij}\right) \times \sin\left(\hat{\alpha}_{\ ij} -\ n_{\hat{\alpha}\ ij}\ \right)\ \times\ cos\ \ \left(\widehat{\varphi}_{\ ij} - n_{\widehat{\varphi}\ ij}\right) \tag{18}$$

By neglecting $n_{\hat{d}\ ij}$, and by using trigonometry:

sin (A - B) = sin A cos B - cos A sin B [12]

Therefore

$$\sin\left(\hat{\alpha}_{\ ij} -\ n_{\hat{\alpha}\ ij}\ \right) = (\ \sin\hat{\alpha}_{\ ij}\ \times \cos n_{\hat{\alpha}\ ij}\ ) - (\ \cos\hat{\alpha}_{\ ij}\ \ \times \sin\ n_{\hat{\alpha}\ ij}\ )$$

Since $n_{\hat{\alpha}\ ij}$ is small number,

Therefore

$$\cos n_{\hat{\alpha}\ ij} \approx 1$$

$$\sin\ n_{\widehat{\alpha}_{ij}} \approx 0$$

By using trigonometry

cos (A - B) = cos A cos B + sin A sin B [12]

Therefore

$$cos\ \left(\widehat{\varphi}_{ij} - n_{\widehat{\varphi}_{ij}}\right) = \left(\cos\widehat{\varphi}_{ij} \times \cos n_{\widehat{\varphi}_{ij}}\right) + \left(\sin\widehat{\varphi}_{ij} \times \sin\ n_{\widehat{\varphi}_{ij}}\right)$$

Since $n_{\widehat{\varphi}_{ij}}$ is small number,

Therefore

$$\cos\ n_{\widehat{\varphi}_{ij}} = 1$$

$$\sin n_{\widehat{\varphi}_{ij}} = 0$$

So that equation (18) can be written as

$$xj - xi \approx \widehat{d}_{ij} \times \sin\widehat{\alpha}_{ij} \times cos\ \widehat{\varphi}_{ij} \qquad (19)$$

Equation (20) is extracted from figure 3 and figure 4

$$yj - yi = \left(\widehat{d}_{ij} - n_{\widehat{d}_{ij}}\right) \times \sin\left(\widehat{\alpha}_{ij} - n_{\widehat{\alpha}_{ij}}\right) \times sin\ \left(\widehat{\varphi}_{ij} - n_{\widehat{\varphi}_{ij}}\right) \qquad (20)$$

By neglecting $n_{\widehat{d}_{ij}}$, and by using trigonometry

sin (A - B) = sin A cos B - cos A sin B [12]

Therefore

$$\sin\left(\widehat{\alpha}_{ij} - n_{\widehat{\alpha}_{ij}}\right) = \left(\sin\widehat{\alpha}_{ij} \times \cos n_{\widehat{\alpha}_{ij}}\right) - \left(\cos\widehat{\alpha}_{ij} \times \sin\ n_{\widehat{\alpha}_{ij}}\right)$$

By neglecting $n_{\widehat{d}_{ij}}$ and by using trigonometry

sin (A - B) = sin A cos B - cos A sin B [12]

Therefore

$$\sin\left(\widehat{\alpha}_{ij} - n_{\widehat{\alpha}_{ij}}\right) = \left(\sin\widehat{\alpha}_{ij} \times \cos n_{\widehat{\alpha}_{ij}}\right) - \left(\cos\widehat{\alpha}_{ij} \times \sin\ n_{\widehat{\alpha}_{ij}}\right)$$

Since $n_{\widehat{\alpha}_{ij}}$ is small number,

Therefore

$$\cos n_{\widehat{\alpha}_{ij}} \approx 1$$

$$\sin\ n_{\widehat{\alpha}_{ij}} \approx 0$$

And

$$\sin\left(\widehat{\varphi}_{ij} - n_{\widehat{\varphi}_{ij}}\right) = \left(\sin\widehat{\varphi}_{ij} \times \cos n_{\widehat{\varphi}_{ij}}\right) - \left(\cos\widehat{\varphi}_{ij} \times \sin\ n_{\widehat{\varphi}_{ij}}\right)$$

Since $n_{\widehat{\varphi}_{ij}}$ is small number,

Therefore

$$\cos n_{\widehat{\varphi}_{ij}} \approx 1$$

$$\sin\ n_{\widehat{\varphi}_{ij}} \approx 0$$

Therefore

$$yj - yi \approx \hat{d}_{ij} \times \sin \hat{\alpha}_{ij} \times sin\, \hat{\varphi}_{ij} \tag{21}$$

From figure 4

$$\cos\ \alpha\, \text{i, j} = \frac{\text{zj} - \text{zi}}{d\ i,j} \tag{22}$$

$$\text{zj} - \text{zi} = d\ i,j \times \cos \alpha\ \text{i, j} \tag{23}$$

If the estimation error is taken into consideration

$$\text{zj} - \text{zi} = \left(\hat{d}_{ij} - n_{\hat{d}_{ij}}\right) \times \cos(\hat{\alpha}_{ij} - n_{\hat{\alpha}_{ij}}) \tag{24}$$

By neglecting $n_{\hat{d}_{ij}}$ and by using trigonometric formulas

cos(A - B) = cos A cos B + sin A sin B [12]

Therefore

$$\cos(\hat{\alpha}_{ij} - n_{\hat{\alpha}_{ij}}) = (\cos \hat{\alpha}_{ij} \times \cos\ n_{\hat{\alpha}_{ij}}) + (\sin \hat{\alpha}_{ij} \times \sin\ n_{\hat{\alpha}_{ij}})$$

Since $n_{\hat{\alpha}_{ij}}$ is small number

Therefore

$$\cos n_{\hat{\alpha}_{ij}} \approx 1$$

$$\sin\ n_{\hat{\alpha}_{ij}} \approx 0$$

Therefore

$$\text{zj} - \text{zi} \approx \hat{d}_{ij} \times \cos \hat{\alpha}_{ij} \tag{25}$$

4. Collision Avoidance Scheme

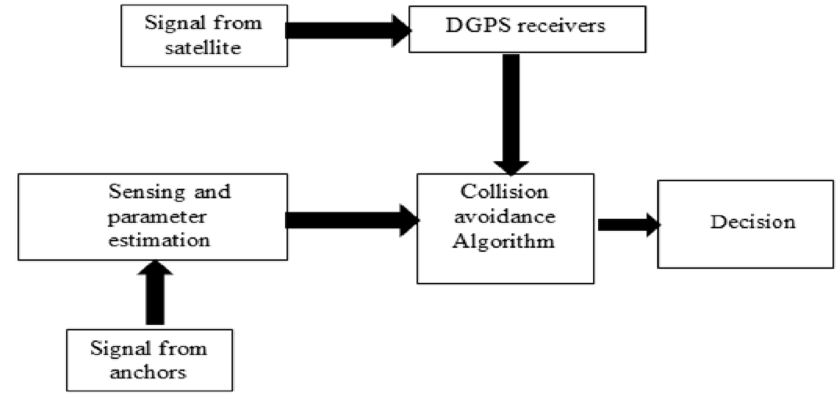


Figure 5. Block diagram of the collision avoidance improvement scheme.

First, the signal is taken from satellite is given to DGPS which is more accurate than GPS. Anchors produce signals used in sensing and estimating parameters. There are some estimates and sensing parameters produced by sensors embedded in the node, and then the collision avoidance algorithm (CSMA/CA) is used to solve the problem of collision avoidance. Finally, the decision can be taken, as shown in figure 5 [13, 14, 15].

5. Simulation

The simulation program is implemented in discrete event simulator, generating WSN which consists of 4 nodes, 8 nodes and 16 nodes. According to the program outputs the nodes will be communicated. The area used in the simulation is $X = 10\text{ m}$ and $Y = 5\text{ m}$. Figure 6 shows four nodes distributed uniformly, the symbol cross indicate for a node. Figure 7 shows 16 nodes in uniform distribution .The simulation needs to run several times for each network topology size 4, size 8and size 16.

In a network topology consisting of four nodes, the program is run 25 times. It produces two networks 1 and 2. Figure 6 shows one of these networks. The distribution of nodes is random during the run process.

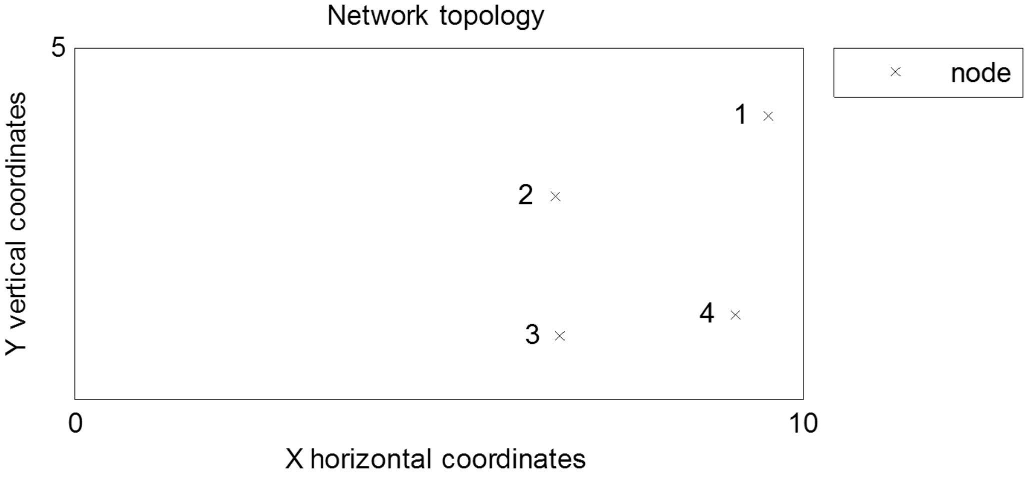


Figure 6. Network topology consisting of 4 nodes.

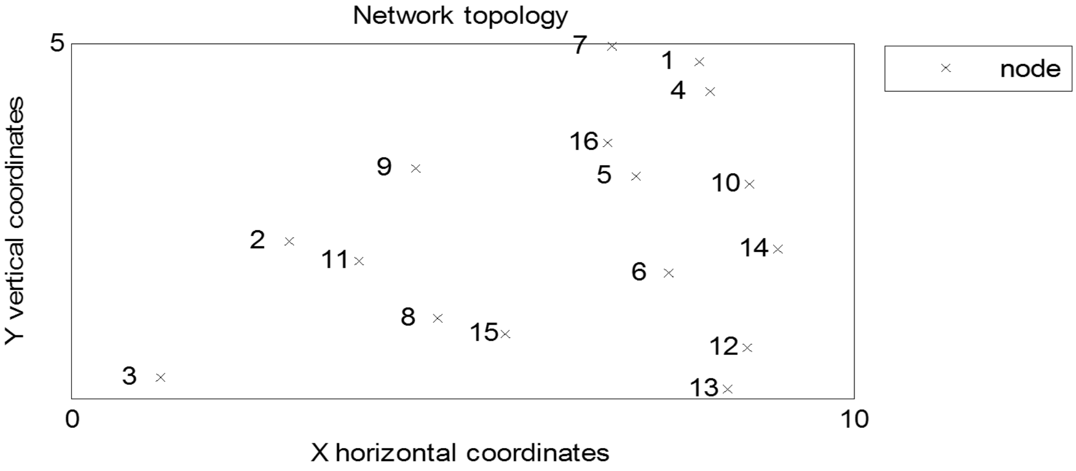


Figure 7. Network topology consisting of 16 nodes.

The source node first senses the channel if busy or idle before sending the data. The channel uses strategy with backoff until the channel is idle. After the channel is found idle, the source node wait specific time which is called Distributed Interframe Space (DIFS) then sends Request to Send (RTS). After receiving (RTS) the source node wait specific time which is called Short Interframe Space (SIFS), then destination node sends Clear to Send (CTS) to the source node.

The source node send data after time equal to SIFS .The destination node sends acknowledgment after waiting time equal to SIFS.

In order to communicate through the network and solve the collision problem, each program starts by sending an application. In send application (send_app) action, there is an initiation of sending data from the source node to the destination node. Receive application action (recv_app) is taken at the destination node, and the node initiating receiving from source node .Send network (send_net) action means that data starts flowing through the network .The program produces two networks, so there is send_net and send_net2. Receive network (recv_net) action which means that received packets are received by destination. In send_mac, if there are waiting packets in mac_que, the node should send them anyway as if the current packet is done at the transmitter. In recv_mac, if there are waiting packets in mac_que, the node should send them anyway as if the current packet is done at the receiver.

In wait for channel action (wait_for_channel) the node senses the channel. If the channel is busy, the node has to wait for a certain time until the channel is free. If the node is not idle, it may be receiving, so (wait_for_channel) action is taken until this receiving is finished or if the channel is not free, it must wait until the channel is free [16].After waiting DIFS, backoff starts. Backoff means that the node waits before resending again after collision, the channel become busy during DIFS. It waits until the channel is free. If the node is still idle and the channel is free, it continues the backoff process, then the source node sends RTS (timeout_rts), after receiving RTS and waiting SIFS, the destination node sends CTS to the source node. The source node then sends data after the SIFS period (timeout_data). When data reaches the destination, acknowledgment is sent to the source, (timeout_rreq) action check if the request is acknowledged or not [17].

Send physical (send_phy) action is taken when the transmitting node switches to transmit mode which broadcasts from source and the destination node. Due to the broadcast nature in wireless channel, every idle node may sense this transmission [18] .Receive physical (recv_phy) action is taken when the receiver switches to receiving mode. Send physical finish action (send_phy_finish) is taken when the destination node finishes receiving packets from the source.

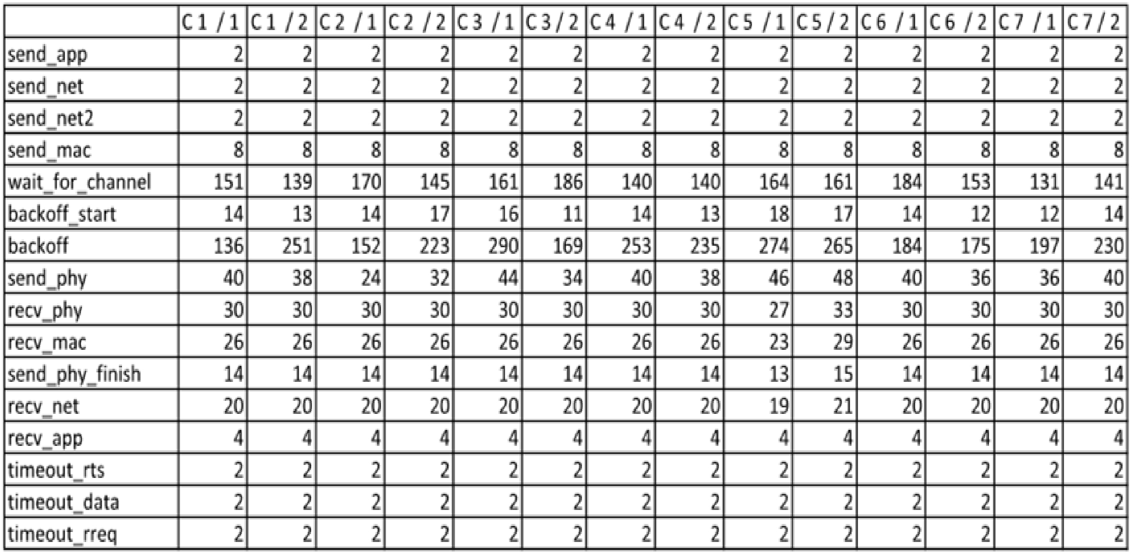
These steps are repeated along the run process several times and the number of the repetition differs among the three network topologies network size 4, network size 8 and network size 16.

6. Results and Discussion

6.1 The first network topology consisting of 4 nodes.

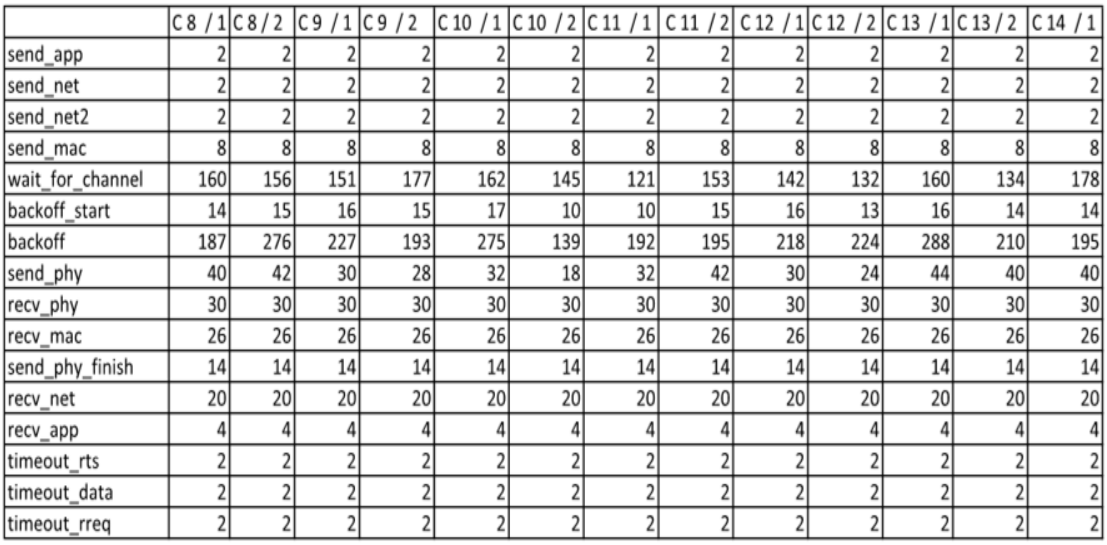
The results for four nodes network are given in table 1.The number of action taken each time is counted during the run process, for example the action send_app is done twice each time. The tables list the number of counts, for example, count 1 is written c1/1 and c1/2 which means that the program produces 2 networks each time the program is run. C1/1 shows the number of commands for network 1 and c1/2 shows number of commands for network 2,c2/1 shows the number of commands for network 1 after running the program second time,c2/2 shows the number of commands for network 2 and etc… as shown in table 1.a.

Table 1.a from count1 (c1/1) to count7 (c7/2).



	C 1 / 1	C 1 / 2	C 2 / 1	C 2 / 2	C 3 / 1	C 3 / 2	C 4 / 1	C 4 / 2	C 5 / 1	C 5 / 2	C 6 / 1	C 6 / 2	C 7 / 1	C 7 / 2
send_app	2	2	2	2	2	2	2	2	2	2	2	2	2	2
send_net	2	2	2	2	2	2	2	2	2	2	2	2	2	2
send_net2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
send_mac	8	8	8	8	8	8	8	8	8	8	8	8	8	8
wait_for_channel	151	139	170	145	161	186	140	140	164	161	184	153	131	141
backoff_start	14	13	14	17	16	11	14	13	18	17	14	12	12	14
backoff	136	251	152	223	290	169	253	235	274	265	184	175	197	230
send_phy	40	38	24	32	44	34	40	38	46	48	40	36	36	40
recv_phy	30	30	30	30	30	30	30	30	27	33	30	30	30	30
recv_mac	26	26	26	26	26	26	26	26	23	29	26	26	26	26
send_phy_finish	14	14	14	14	14	14	14	14	13	15	14	14	14	14
recv_net	20	20	20	20	20	20	20	20	19	21	20	20	20	20
recv_app	4	4	4	4	4	4	4	4	4	4	4	4	4	4
timeout_rts	2	2	2	2	2	2	2	2	2	2	2	2	2	2
timeout_data	2	2	2	2	2	2	2	2	2	2	2	2	2	2
timeout_rreq	2	2	2	2	2	2	2	2	2	2	2	2	2	2

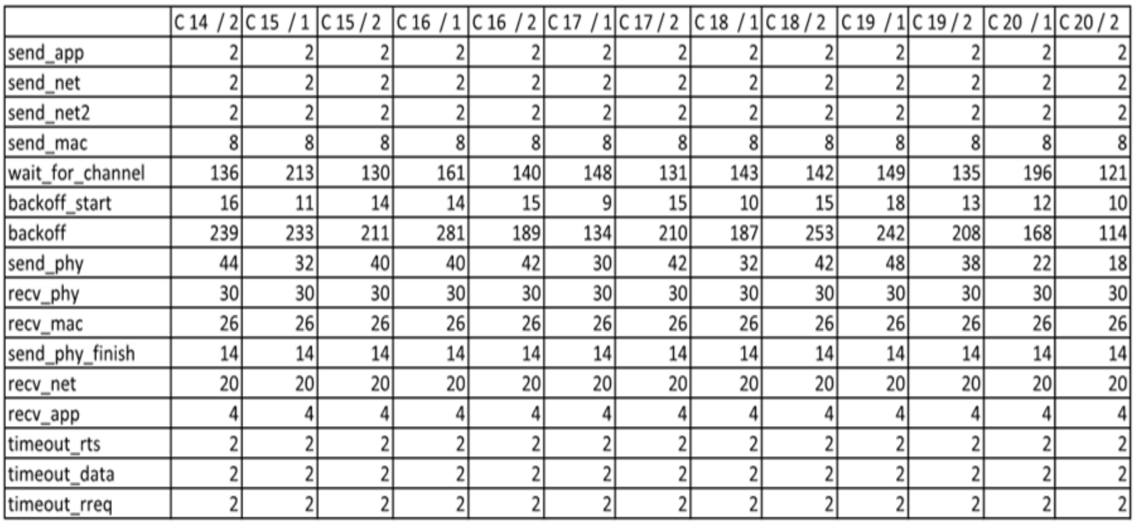
Table 1.b from count8 (c8/1) to count14 (c14/1).



	C 8 / 1	C 8 / 2	C 9 / 1	C 9 / 2	C 10 / 1	C 10 / 2	C 11 / 1	C 11 / 2	C 12 / 1	C 12 / 2	C 13 / 1	C 13 / 2	C 14 / 1
send_app	2	2	2	2	2	2	2	2	2	2	2	2	2
send_net	2	2	2	2	2	2	2	2	2	2	2	2	2
send_net2	2	2	2	2	2	2	2	2	2	2	2	2	2
send_mac	8	8	8	8	8	8	8	8	8	8	8	8	8
wait_for_channel	160	156	151	177	162	145	121	153	142	132	160	134	178
backoff_start	14	15	16	15	17	10	10	15	16	13	16	14	14
backoff	187	276	227	193	275	139	192	195	218	224	288	210	195
send_phy	40	42	30	28	32	18	32	42	30	24	44	40	40
recv_phy	30	30	30	30	30	30	30	30	30	30	30	30	30
recv_mac	26	26	26	26	26	26	26	26	26	26	26	26	26
send_phy_finish	14	14	14	14	14	14	14	14	14	14	14	14	14
recv_net	20	20	20	20	20	20	20	20	20	20	20	20	20
recv_app	4	4	4	4	4	4	4	4	4	4	4	4	4
timeout_rts	2	2	2	2	2	2	2	2	2	2	2	2	2
timeout_data	2	2	2	2	2	2	2	2	2	2	2	2	2
timeout_rreq	2	2	2	2	2	2	2	2	2	2	2	2	2

Table 1.b shows the count number of running process of the program from c8/1 to c14/1.C8/1 shows the number of actions have been taken in the run number eight, as mentioned before the program produces two networks.C8/1 show actions taken for network number 1, c8/2 shows actions taken for network number 2 till c14/1 which show the actions taken for network number one when the run process happen fourteenth time.

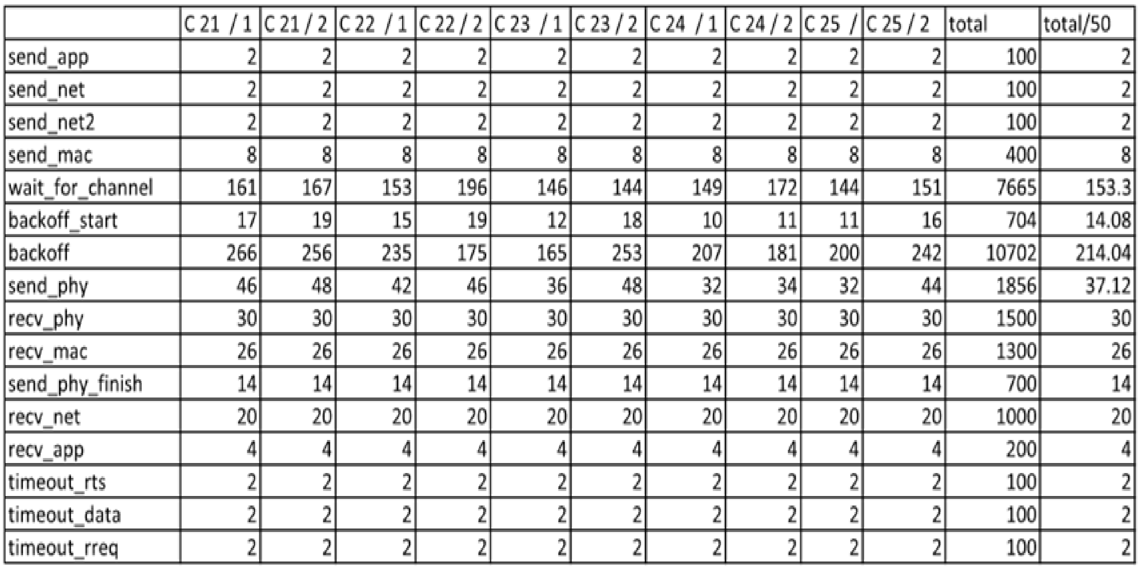
Table 1.c from count14 (c14/2) to count20 (c20/2).



	C 14 / 2	C 15 / 1	C 15 / 2	C 16 / 1	C 16 / 2	C 17 / 1	C 17 / 2	C 18 / 1	C 18 / 2	C 19 / 1	C 19 / 2	C 20 / 1	C 20 / 2
send_app	2	2	2	2	2	2	2	2	2	2	2	2	2
send_net	2	2	2	2	2	2	2	2	2	2	2	2	2
send_net2	2	2	2	2	2	2	2	2	2	2	2	2	2
send_mac	8	8	8	8	8	8	8	8	8	8	8	8	8
wait_for_channel	136	213	130	161	140	148	131	143	142	149	135	196	121
backoff_start	16	11	14	14	15	9	15	10	15	18	13	12	10
backoff	239	233	211	281	189	134	210	187	253	242	208	168	114
send_phy	44	32	40	40	42	30	42	32	42	48	38	22	18
recv_phy	30	30	30	30	30	30	30	30	30	30	30	30	30
recv_mac	26	26	26	26	26	26	26	26	26	26	26	26	26
send_phy_finish	14	14	14	14	14	14	14	14	14	14	14	14	14
recv_net	20	20	20	20	20	20	20	20	20	20	20	20	20
recv_app	4	4	4	4	4	4	4	4	4	4	4	4	4
timeout_rts	2	2	2	2	2	2	2	2	2	2	2	2	2
timeout_data	2	2	2	2	2	2	2	2	2	2	2	2	2
timeout_rreq	2	2	2	2	2	2	2	2	2	2	2	2	2

Table 1.c continues showing the results produced each time the run process is done .Table 1.c begins from c14/2 which shows the actions taken for network number 2 in the run number fourteen .Then the other columns show the rest of the results until c20/ 2, c20/2 mentioned in the last column shows actions taken when the run process is done number twenty, c20/1 shows results for network number 1 and c20/2 shows results for network number 2.

Table 1.d from count21 (c21/1) to count25 (c25/2).



	C 21 / 1	C 21 / 2	C 22 / 1	C 22 / 2	C 23 / 1	C 23 / 2	C 24 / 1	C 24 / 2	C 25 /	C 25 / 2	total	total/50
send_app	2	2	2	2	2	2	2	2	2	2	100	2
send_net	2	2	2	2	2	2	2	2	2	2	100	2
send_net2	2	2	2	2	2	2	2	2	2	2	100	2
send_mac	8	8	8	8	8	8	8	8	8	8	400	8
wait_for_channel	161	167	153	196	146	144	149	172	144	151	7665	153.3
backoff_start	17	19	15	19	12	18	10	11	11	16	704	14.08
backoff	266	256	235	175	165	253	207	181	200	242	10702	214.04
send_phy	46	48	42	46	36	48	32	34	32	44	1856	37.12
recv_phy	30	30	30	30	30	30	30	30	30	30	1500	30
recv_mac	26	26	26	26	26	26	26	26	26	26	1300	26
send_phy_finish	14	14	14	14	14	14	14	14	14	14	700	14
recv_net	20	20	20	20	20	20	20	20	20	20	1000	20
recv_app	4	4	4	4	4	4	4	4	4	4	200	4
timeout_rts	2	2	2	2	2	2	2	2	2	2	100	2
timeout_data	2	2	2	2	2	2	2	2	2	2	100	2
timeout_rreq	2	2	2	2	2	2	2	2	2	2	100	2

Table 1.d shows the count number of running the program from c21/1 to c25/2.C21/1 shows the number of actions taken in the run number twenty one, as mentioned before the program produces two networks, c21/1 show actions taken for network number 1, c21/2 shows actions taken for network number 2 until c25/2 which show the actions taken for network number 2 when the run process is done twenty fifth time. The last two columns contain the total number of actions done (total) and the average number of repetition (total/50).

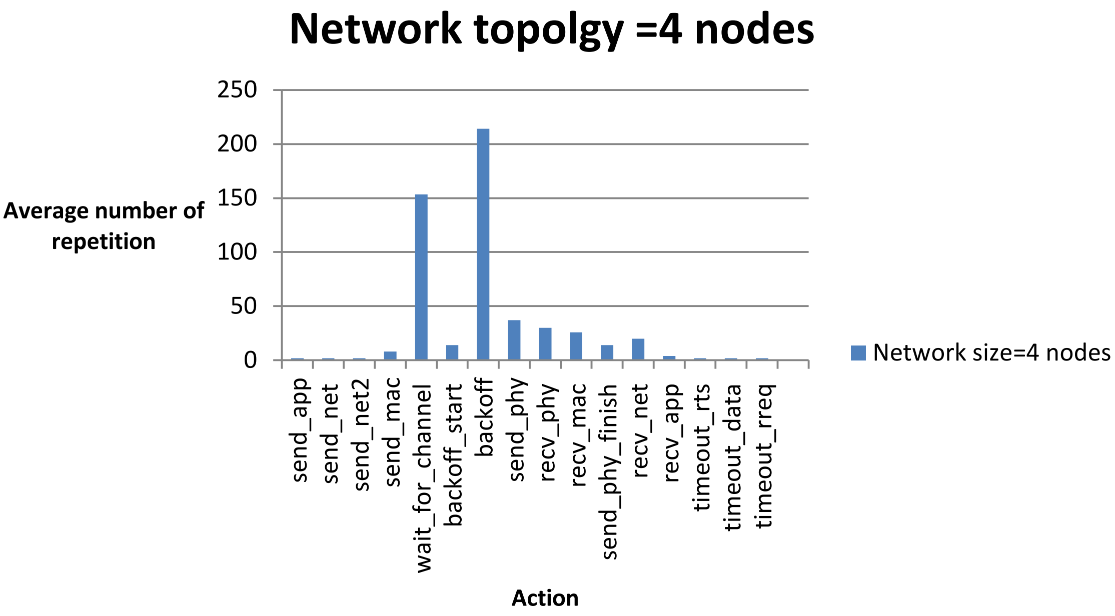


Figure . 8. Results of 4 nodes network.

Table 1.e Average number for each action for 4 nodes.

Action	Average number of repetition
send_app	2
send_net	2
send_net2	2
send_mac	8
wait_for_channel	153.3
backoff_start	14.08
Backoff	214.04
send_phy	37.12
recv_phy	30
recv_mac	26
send_phy_finish	14
recv_net	20
recv_app	4
timeout_rts	2
timeout_data	2
timeout_rreq	2

Figure 8 explains table 1.e that shows the average number of repetition of each action taken send_app, send_net, send_net2, timeout_rts, timeout_data, timeout_rreq average number is 2, send _mac average number is 8, wait_for_channel average number is 153.3, backoff_start average number is 14.08, backoff average number is 214.04, send_phy average number is 37.12, recv_phy average number is 30, recv_mac average number is 26 and send_phy_finish average number is 14, recv_net average number is 20, recv_app average number is 4.

6.2 The second network topology consists of 8 nodes.

The second topology based on 8 nodes is built and the results are presented in figure 9 and table 2.

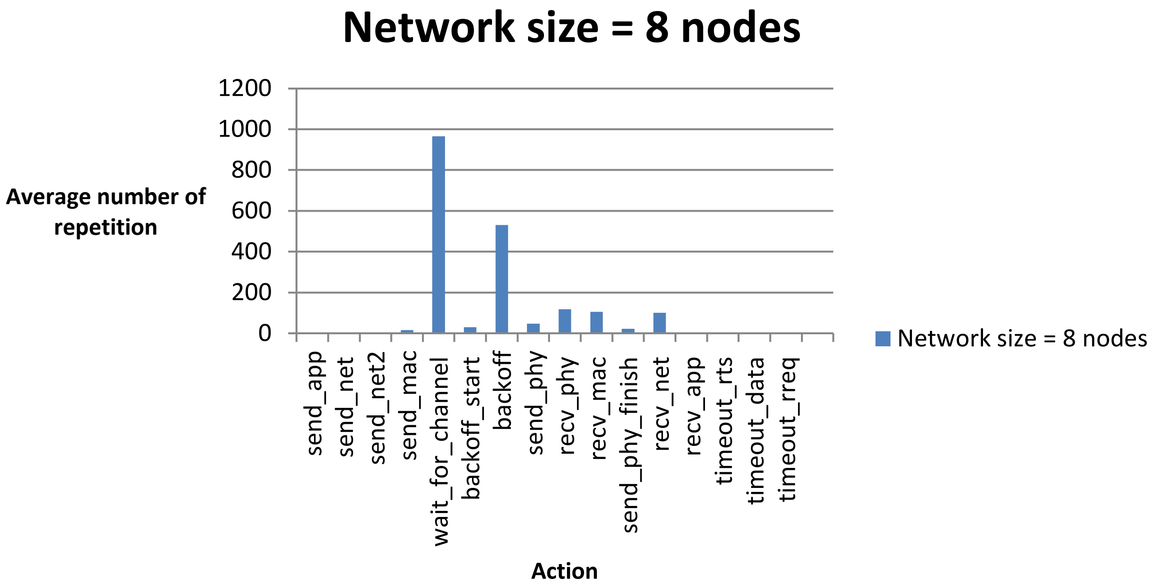


Figure 9. Results of 8 nodes network.

Table 2. Average number for each action for 8 nodes.

Action	Average number of repetition
send_app	2
send_net	2
send_net2	2
send_mac	16
wait_for_channel	964.92
backoff_start	30.54
Backoff	530.78
send_phy	46.88
recv_phy	118
recv_mac	106
send_phy_finish	22
recv_net	100
recv_app	4
timeout_rts	2
timeout_data	2
timeout_rreq	2

Figure 9 explains table 2 that shows the average number of repetition of each action taken send_app, send_net, send_net2, timeout_rts, timeout_data, timeout_rreq average number is 2, send _mac average number is 16, wait_for_channel average number is 964.92, backoff_start average number is 30.54, backoff average number is 530.78, send_phy average number is 46.88, recv_phy average number is 118, recv_mac average number is 106 and send_phy_finish average number is 22. recv_net average number is 100, recv_app average number is 4.

6.3 The third network topology consisting of 16 nodes.

The third topology based on 16 nodes is built and the results are presented in figure 10 and table 3.

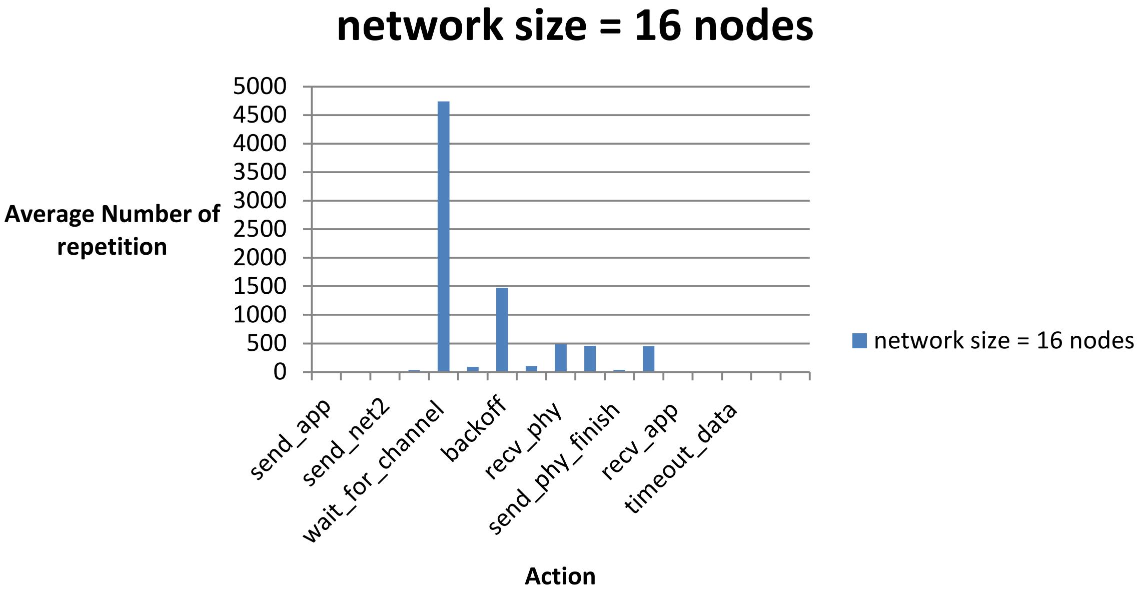


Figure 10. Results of 16 nodes network.

Table 3. Average number for each action for 16 nodes.

Action	Average number of repetition
send_app	2
send_net	2
send_net2	2
send_mac	32
wait_for_channel	4737.84
backoff_start	85.36
Backoff	1471.66
send_phy	106.2
recv_phy	486
recv_mac	458
send_phy_finish	38
recv_net	452
recv_app	4
timeout_rts	2
timeout_data	2
timeout_rreq	2

Figure 10 explains table 3 that shows the average number of repetition of each action taken send_app, send_net, send_net2, timeout_rts, timeout_data, timeout_rreq average number is 2, send _mac average number is 32, wait_for_channel average number is 4737.84, backoff_start average number is 85.36, backoff average number is 1471.66, send_phy average number is 106.2, recv_phy average number is 486, recv_mac average number is 458 , send_phy_finish average number is 38, recv_net average number is 452, recv_app average number is 4.

7. Conclusion and Future work

In this paper three topologies are introduced to build WSN based on 4 nodes, 8 nodes, and 16 nodes .The localization and collision avoidance of transfer packets are simulated and several results are extracted. It is found that DGPS is preferable than GPS due to accurate detection of location, and less percentage error. In relation to collision and according to simulation results ,increasing the number of nodes will increase the collision problem. The average number of send_app, send_net, send_net2, recv_app, timeout_rts, timeout_data and timeout_rreq is the same for the three network topologies, but the average number of send_mac, wait_ for _channel, backoff_start, backoff, send_phy, recv_phy, recv_mac,send_phy_finish and recv_net is increasing as the number of nodes increases. It is concluded that implementation of topologies under LOS condition is preferable than NLOS, due to less time consuming and less complexity. For future work collision avoidance and localization can be studied for different distributions, such as, gaussian distribution, rayleigh distribution, circular distribution.

References

[1] Kegen Yu & Y. Jay Guo (2009) “Anchor Global Position Accuracy Enhancement Based on Data Fusion” IEEE transactions on vehicular technology Vol. 57, No. 3, pp 1616-1623.

[2] Nirupama Bulusu, John Heidemann, & Deborah Estrin (2000) “GPS-less Low Cost Outdoor Localization for Very Small Devices” IEEE Personal Communications Magazine, Vol. 7, No.5, pp 28-34.

[3] Seyedeh Zahra Sadri, Tabaee Zavareh & Mahmood Fathy (2012) “ Rate Allocation for SVC transmission in Energy-Constrained Wireless Sensor Networks” International Journal of Computer Networks & Communications (IJCNC), Vol. 4, No.1 , pp127-145.

[4] Po-Hsuan Tseng, Student Member, & Kai-Ten Feng (2009) “Hybrid Network/Satellite-Based Location Estimationand Tracking Systems for Wireless Networks” transactions on vehicular technology Vol. 58, No. 9, pp 5174-5189.

[5] Lucianaz C, Rorato O, Allegretti M, Mamino M, Roggero M & Diotri F (2011) “Low cost DGPS wireless network”, IEEE-APS Topical Conference on Antennas and Propagation in Wireless Communications, Torino, Vol.4 , pp 12-16.

[6] Behrouz A.Forouzan, & Sophia Chung Fegan (2004) “Data communication and networking”3rd edition, ISBN 007-123241-9

[7] Wei Ye, John S. Heidemann, & Deborah Estrin (2004) “Medium access control with coordinated adaptive sleeping for wireless sensor networks” IEEE/ACM Transactions on Networking, Vol.12, No.3, pp 493–506.

[8] Shweta Singh & Ravindara Bhatt (2012) “Adjacency Matrix Based Energy Efficient Scheduling Using S-Mac Protocol in Wireless Sensor Networks “International Journal of Computer Networks & Communications (IJCNC), Vol. 4, No2, pp 149-168.

[9] Kamal Kumar Sharma, Ram Bahadur Patel & Harbhajan Singh (2010) “A Reliable and Energy Efficient Transport Protocol for Wireless Sensor Networks “International Journal of Computer Networks & Communications (IJCNC), Vol.2, No 5, pp 92-103.

[10] John A. Stankovic (2006) “Wireless Sensor Networks” University of Virginia,Charlottesville, Virginia.

[11] K. Yu & Y. J. Guo (2008) “Improved positioning algorithms for non-line-of-sight environments” IEEE Trans. Veh. Technol, Vol. 57, No. 4, pp 2342–2353.

[12] Earl Swokowski & Jeffery A. Cole (2010) “Algebra and Trigonometry with Analytic Geometry, Classic Edition”12 th edition, Thomson Brooks/Cole publisher, ISBN: 0495559717.

[13] Thanos Stathopoulos, Rahul Kapur, Deborah Estrin, John Heidemann, & Lixia Zhang (2004) “Application-Based Collision Avoidance in Wireless Sensor Networks” 29th IEEE International Conference on Local Computer Networks, pp. 506-514.

[14] J.H.Wang & Y.Gao (2007) “High-sensitivity GPS data classification based on signal degradation conditions” *IEEE Trans. Veh. Techno*, Vol. 56, No. 2, pp. 566–574.

[15] A. S. Zaidi & M. R. Suddle (2006) “Global navigation satellite systems” *Int. Conf. Advances Space Techno*, pp. 84–87.

[16] W. Ye, J. Heidemann & D. Estrin (2002) “An energy-efficient mac protocol for wireless sensor networks” IEEE INFOCOM, New York, pp. 1567 –1576.

[17] Ravi Tandon (2012) “Determination of Optimal Number of Clusters in Wireless Sensor Networks”International Journal of Computer Networks & Communications (IJCNC), Vol. 4, No. 4, pp 235-249.

[18] Mohammad Waris Abdullah & Nazar Waheed (2012) “Performance and Detection of M-ary Frequency Shift Keying in Triple Layer Wireless Sensor Network” International Journal of Computer Networks & Communications (IJCNC), Vol. 4, No. 4, pp 177- 189.