# Performance Comparison and Node Failure Assessment of Energy Efficient Two Level Balanced and Progressive Sensor Networks


Maneesha V Ramesh
Amrita Center for Wireless Networks and Applications
Amrita Vishwa Vidyapeetham
Kerala, India
maneesha@am.amrita.edu

Sreedevi A. G.
Amrita Center for Wireless Networks and Applications
Amrita Vishwa Vidyapeetham
Kerala, India
agsreedevi@gmail.com

Kamalanathan Kandasamy
Amrita Center for Wireless Networks and Applications
Amrita Vishwa Vidyapeetham
Kerala, India
kamalanathan@am.amrita.edu

P Venkat Rangan
Vice Chancellor
Amrita University
India
venkat@amrita.edu



## ABSTRACT

This research is concerned with the design of Tree WSNs. This research proves that configuration plays a vital role in energy optimisation, node failure and network lifetime when designing a Tree WSN. We compare a Progressive Two Level Tree WSN with a Balanced Two Level Tree WSN. Our simulation results prove that a Progressive configuration has two advantages, over a Balanced configuration, which are less computations required to complete each process and more tolerance of node failures. These Progressive configuration advantages both lead to more energy efficiency compared to a Balanced configuration. Therefore, the lifetime of a Progressive Two Level Tree WSN is longer than that of a Balanced Two Level Tree WSN.



## Categories and Subject Descriptors

C.4 [Performance of Systems]

## General Terms

Algorithm, Performance, Experimentation

## Keywords

Wireless Sensor Networks (WSN), Two Level Balanced Network, Two Level Progressive Network, Node Failure, Network Lifetime, and Energy Consumption.




## 1. INTRODUCTION

In most applications, Wireless Sensor Networks (WSN) have proved to be an effective technology for monitoring, detecting, estimating the probability of occurrence, and issuing early warnings. Minimizing the energy consumption for the sustained operation of each network is of the highest priority.

WSNs allow the observation and monitoring of previously unobservable phenomena. WSNs consist of a system of distributed sensors implemented in the physical world. Since these sensor nodes are exposed to harsh and unpredictable environments, sensor network applications must be capable of handling a wide variety of faults such as software errors, node and link failures, and network partitions. Building a reliable and robust sensor network system is a tedious challenge.

There are many protocols and algorithms available, which can be chosen based according to the nature of the applications. This will conserve energy at different levels. Various factors such as the lifetime, the utilization of network, the packet delays, the interference, the robustness, the mobility, the failure and duty cycle, the directional antennas, and the transmission power decide the selection of the protocol. Considering all these factors, we compare two different configurations (progressive and balanced) for a two level tree structured WSN. The results of the comparison were used to develop power optimization strategies.

In our study, we consider the impact that WSN configuration has on its power consumption. In a WSN tree structure, there is more than one level, (in fact the tree structure is the only WSN structure to have two or more levels).In this study, we compare a balanced two level tree WSN with a progressive two level tree WSN. The design of a progressive tree structure differs from a balanced tree structure due its configuration that is the connections of its Sensor Nodes (SNs) to its Intermediate Nodes (INs).

We hypothesise that if we carefully adjust the configuration of atree WSN, thenwe can reduce the energy consumed by the whole WSN. The concept backing our hypothesis is that as the number of computations and transmissions is reduced, the energy consumed by the processing and the radio system will also reduce, whichin turn reduces the overall energy consumed by the WSN.

As well as WSN configuration, power consumption has also been linked to node failure. After node fails the power consumption of the network will increase as the other nodes have to take on the work of the failed node .Node failures occur much more frequently in wireless networks compared to wired networks. Hardware degradation, inaccurate readings, and environmental changes all at to the risk of WSN failure. Node failure decreases the distributed detecting performance of a system. We discuss node failures on different levels that are possible in case of the balanced and progressive two level tree WSNs and propose solutions.

The rest of the paper is organized as follows: A literature survey is presented in section 2. Two level Balanced and Progressive sensor network topologies and their energy optimization are described in section 3. In sections 4 to 6, we analyse node failure handling strategies in different levels of the Balanced and Progressive two level tree WSNs and propose solutions. Section 7 contains results and observations. Finally we conclude in section 8.

## 2. RELATED WORK

Power optimisation in WSNs has been undertaken in various ways. Duty cycle control algorithms conserved power, but compromised the efficiency of the network in computation and prediction [1, 2]. The response to the duty cycle control problem was to introduce topology control algorithms where appropriate power consumption was tailored to the appropriate network structure [3, 4, 5], be it: random, star, tree or multi-hop etc.

In a WSN tree structure, there is more than one level, (in fact the tree structure is the only WSN structure to have two or more levels). We focus on progressive two level tree WSN and balanced two level tree WSN power optimisation solutions. A key factor that increases WSN power consumption is node failure. When node failures increase power consumption also increases [6].

Limited studies have been conducted on solutions to node failure specifically in tree WSNs. One such paper, analysed the effect of node failures and unreliable communications in a dense sensor network, arranged as a tree of bounded height [7]. [7] analysed and discussed the effects of tree WSN node failures; whereas we look at tree WSN power optimising, node failure identification and recovery solutions. Node failure, identification and recovery will differ dependent on the node's level in that tree hierarchy. Our work in this paper analyses different solutions available for a tree WSN, when a node fails.

## 3. BALANCED AND PROGRESSIVE WSN HIERARCHY

In our study, we consider the impact that WSN configuration has on its power consumption. WSNs can have hierarchical tree structure.In a WSN tree structure, there is more than one level, (in fact the tree structure is the only WSN structure to have two or more levels). For the sake of clarity, we define our tree structure levels as the Gateway Node (GN) being on level 0; the Intermediate Nodes (IN) being on level 1; and the Sensor Nodes (SN) being on level 2.In this study, we consider WSNs tree structures of only two levels (Levels 0, 1, 2).Sensor Nodes (SNs) have sensors attached; SNs sense the data and transmit it to the Intermediate Nodes (INs). INs aggregate, compute(finding either the average or maximum or minimum from the received values),and transmit data to the Gateway Node (GN).

In this study, we compare a balanced tree WSN with a progressive tree WSN. The design of a progressive tree WSN differs from a balanced tree WSN due its configuration that is the connections of its Sensor Nodes (SNs) to its Intermediate Nodes (INs). Both a progressive tree WSN and a balanced tree WSN are constructed with a set number of SNs and INs. In a balanced tree WSN, SNs are divided equally among the INs. The design of a progressive tree WSN differs from a balanced tree WSN due the connections of its SNs to its INs. A progressive tree WSN, an IN can have a varied number of SNs attached to it, whereas a balanced tree WSN, an IN will have an equal number of SNs attached to it. In the construction of a progressive tree WSN, the number of connections between SNs and INs and how the connections are constructed depend on how many sensors are required in a particular application.

A progressive tree WSN is constructed with the first two INs having the same amount of SNs attached, after that, each IN will have one additional attached SN than the previous IN. This progressive tree WSN configuration is designed to prevent collisions. The GN starts the pipeline operation as soon as it gets the packets from the first two INs. The packet from the third IN, will have a delay pipelined equal to the computation time required to process the first two packets.

We hypothesise that if we carefully adjust the configuration of a tree WSN, then we can reduce the energy consumed by the whole WSN. The concept backing our hypothesis is that as the number of computations and transmissions is reduced, the energy consumed by the processing and the radio system will also reduce, which in turn reduces the overall energy consumed by the WSN.

For each IN there will be a particular amount of SNs. And assuming: for one IN it has n of SNs. In such a situation: while the IN is processing it will need to perform n-1computations. This is the main idea used to formulate the other equations.

**Table 1:The symbols used in the paper.**

| Symbol | Description |
|---|---|
| N | Total number of sensor nodes in the network |
| M | Total number of intermediate nodes on the network |
| $m_{max}$ | The count of intermediate nodes for which the network yields minimum delay |
| Tcomp | Time taken for comparing two sensor values |
| Tnet | Transmission time from one wireless node to another in the WSN |
| Tmax | Total time taken by the WSN for computing the maximum value among all of its sensor values |
| Rs | Sensing radius of a sensor node |

## 3.1 Balanced Two Level Tree WSN

In a Balanced Two Level Tree WSN, the total amount of SNs is divided equally among the INs. Given N of SNs, and m of INs, for a balanced tree WSN, each IN will have N/m of SNs as their children.

The total energy consumed by the GN is proportional to the amount of computations and amount of transmissions. The total delay or energy consumed or time taken by the GN to complete the aggregate computation in a balanced two level tree WSN is,

$$Tmax = \left(\frac{N}{m} - 1\right) Tcomp + (m-1) Tcomp + 2 Tnet \tag{1}$$

The different values of m can give different delays for the WSN. The minimum delay for the WSN is obtained for $m = \sqrt{N}$. Then each IN will have $\sqrt{N}$ of SNs attached to it. The delay of a Balanced Two Level WSN can be calculated using,

$$Tmax = 2(\sqrt{N} - 1)\ Tcomp + 2\ Tnet \tag{2}$$

## 3.2 Progressive Two Level Tree WSN

In a Progressive Two Level Tree WSN, an alternating configuration with the data being sent from the SNs is pipelined to prevent collisions. The GN starts the pipeline operation as soon as it gets the packets from the first two INs. The packet from the third IN will have a delay pipelined equal to the computation time required to process the first two packets.

Let $n1, n2, n3, \dots nm$ be the amount of SNs under INs: $IN1, IN2, IN3, \dots INm$ respectively, in a Progressive Two Level Tree WSN with N of SNs and m of INs. The first two sub trees will have $n1 = n2$ since the data values from the first two sub trees can arrive at the same time at the GN. The third sub tree will have $n3 = n2 + 1 = n1 + 1$ so that the third data value will arrive just when the GN has completed the aggregation of the first two values. Continuing in this manner, each sub tree has a one SN more than its preceding sub tree; the $m^{th}$ (last sub tree) will have $nm = n1 + m - 2$ nodes.

Here also total amount of computations and transmissions at the GN can be calculated. This is observed to give minimal computations when compared with a Balanced Two Level Tree WSN. Here, the total delay for aggregation or max computation is given by,

$$Tmax = (n1 + m - 2)\ Tcomp + 2\ Tnet \tag{3}$$

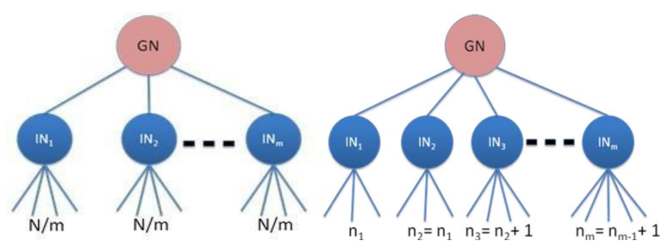
Where n1 is calculated using the relation given below.

$$n1 = \left(\frac{N-1}{m}\right) + \left(\frac{3-m}{2}\right) \tag{4}$$

## 4. NODE FAILURES

The Two Level Tree WSNs, discussed in this paper, has 3 types of nodes working in different levels (SN, IN, GN). Here we will address the node failures impact on energy consumption and the lifetime of the network. The issues addressed in the following sections are:

i. Sensor Node (SN) failure
ii. Intermediate Node (IN) failure




(a) Balanced Network (b) Progressive Network

Figure 1: Two Level Sensor Networks

## 4.1 Assumptions

The solutions proposed for addressing the node failures at different levels are based on the following assumptions.

1. All SNs are time synchronized.
2. INs perform data aggregation, computations and transmissions, no sensing.
3. Mobility is limited to one hop.
4. Energy consumed for one-hop mobility is negligible compared to that of computation and transmission.
5. The SNs have uniform coverage.
6. All the graphs at section 7 are drawn with the m value that gives minimum delay.

# 5. SENSOR NODE (SN) FAILURE

In harsh environments, the probability of SNs failing is high and consequently the data might be lost. Here, we discuss a data recovery protocol which will collect data from the SN failed regions with the help of the mobility of adjacent SNs, where the mobility is limited to one hop. This paper explains the method with respect to the Progressive Two Level Tree WSN. The same method is applied to the Balanced Two Level Tree WSN also.

Figure 2: Rs is the sensing radius of a node. The dotted lines indicate the range of the SN. In figure 2, when one SN fails to collect data from its region, an adjacent mobile SN will collect the failed SN's data and its own data simultaneously. The sensing range of the two neighbouring SNs, intersects at 'a' and 'b'. Vectors: 'Va' and 'Vb' are drawn and by using the triangle rule, the resultant: 'Vr' is found out. The mobile SN:'S' can move in a '–Vr' direction, and 'S' collects data from the failed SN region.

The distance 'S' can travel in '–Vr' direction, without 'S' losing range and data is calculated by the following: Find out the nearest point 'a' or 'b' from 'S'. Let it be 'a' (in this calculation). From 'a' draw a line perpendicular to the resultant intersecting at 'c'. Calculate the distance 'Sc' and mark the distance in the '–Vr' direction.

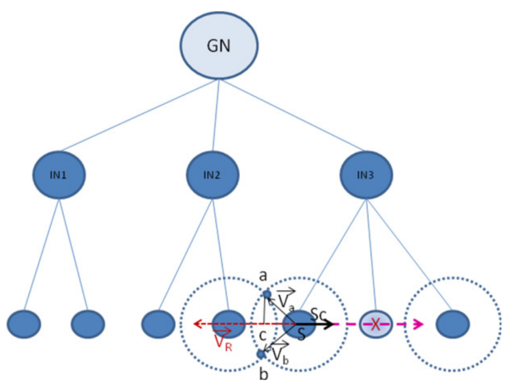



Figure 2: Progressive Two Level Tree WSN before SN Failure

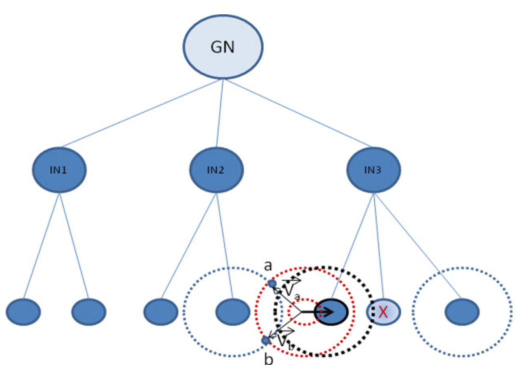



Figure 3: Progressive Two Level Tree WSN after SN Failure

The 'adjacent mobile SN' can move a distance of 'Sc' in '–Vr' direction without losing range and its data. The new range and location of the 'adjacent mobile SN' is shown in Figure 3. The new range of the 'adjacent mobile SN' is obtained is still able to hold the minimum overlap required for communication between two adjacent SNs.

Assuming that there always are two working SNs between two failed SNs, and likewise no two adjacent SNs fail, we will generalize the amount of failures tolerable by an IN. This can be summarised as: For every three SNs, one SN can fail.

## 5.1 Balanced Two Level Tree WSN

Amount of SN failures tolerable for one IN,

$$f = \lfloor \sqrt{N}/3 \rfloor$$

where $\sqrt{N}$ is the amount of SNs attached to one IN.

Amount of SN failures tolerable to the whole BalancedWSN with IN1,IN2,....INmas Intermediate Nodes (INs), where m=$\sqrt{N}$ for minimum delay

F = sum of failures tolerable in each IN nodes

$$= \sqrt{N}\ \lfloor \sqrt{N}/3 \rfloor \quad (5)$$

This will not increase the minimum delay incurred nor the amount of computations at the GN, since the energy for one hop mobility is assumed to be negligible.

## 5.2 Progressive Two Level Tree WSN

Amount of SN failures tolerable for one IN,

$$f = \lfloor n/3 \rfloor$$

where n is the amount of SNs attached to one IN.

Amount of SN failures tolerable to the whole Progressive WSN with IN1,IN2,....INmas Intermediate Nodes (INs),

F = sum of failures tolerable in each IN

$$= \lfloor n1/3 \rfloor + \lfloor n2/3 \rfloor + \ldots + \lfloor n_m/3 \rfloor \quad (6)$$

This will not increase the minimum delay incurred nor the amount of computations at the GN, since the energy for one hop mobility is assumed to be negligible.

# 6. INTERMEDIATE NODE FAILURE

Intermediate Nodes (INs) are nodes at level 1 in the Balanced and Progressive Two Level Tree WSN. INs need to receive data from SNs, and perform data aggressive computations, and then transmit the result to the GN. As Ins perform so many computations, INs have more chance of failing than SNs. IN failure results in SN data not reaching the GN. Since data from SNs is the whole point for the network, we propose a self-reorganisation algorithm for both the Balanced and Progressive Two Level Tree WSN.

## 6.1 Balanced Two Level Tree WSN Self-Reorganisation

Consider a Two Level Tree WSN with N of SNs, m of INs, and N/m of SN to each IN. We assume that any extra SNs will be added one by one to each IN, starting from last IN to first IN. These additional SNs will contribute 1Tcomp additional to the Tmax mentioned in section 3.

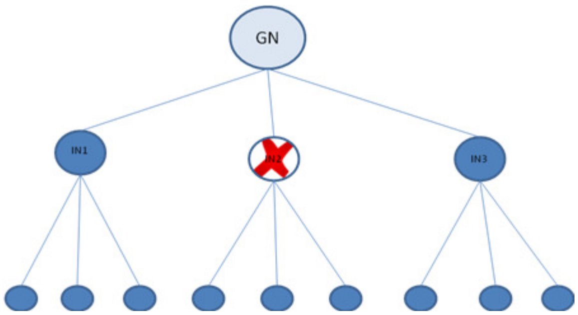



Figure 4 : Balanced Two Level Tree WSN before IN Failure

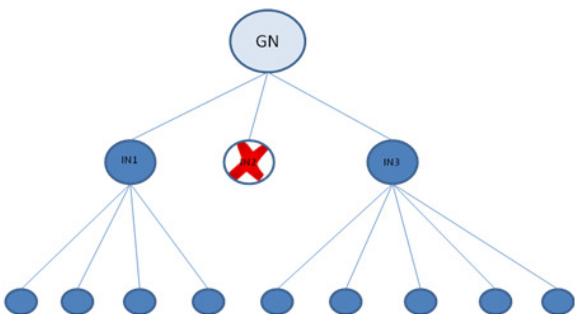



Figure 5: Balanced Two Level Tree WSN after IN Failure and Reorganization

When an IN fails, the WSN executes a self-reorganization algorithm. The SNs will organize themselves using the existing INs. For one IN failure, the WSN parameters changes as: N of SNs, m-1 of INs, and N/(m-1) of SNs to each IN. This new self-organisation will increase the amount of computations involved at each IN, there by leading to a small increase in overall energy consumed.

Consider a Balanced Two Level Tree WSN with $\sqrt{N}$ amount of IN and N/$\sqrt{N}$ of SNs attached to each IN. A Balanced Two Level Tree WSN will have,

$$Tmax = \left(\frac{N}{\sqrt{N}} - 1\right)Tcomp + \left(\sqrt{N} - 1\right)Tcomp + 2Tnet$$

When one IN fails, the new amount of INs will be ($\sqrt{N} - 1$ ). Now the new distribution will have $\frac{N}{\sqrt{N}-1}$ SNs at each IN.

$$Tmax' = \left(\frac{N}{\sqrt{N} - 1} - 1\right)Tcomp + \left(\sqrt{N} - 1 - 1\right)Tcomp + 2Tnet$$

When ‘f’ INs fail,

$$Tmax^f = \left(\frac{N}{\sqrt{N} - f} - 1\right)Tcomp + \left(\sqrt{N} - f - 1\right)Tcomp + 2Tnet \quad (7)$$

**Theorem 1**: The energy consumption of a Balanced Two Level Tree WSNincreases as the amount of IN failures increase. For the successful operation of the network we must tolerate IN failures, however the network life time will be compromised.

*Proof*: Let us assume that the amount of IN failures that the Balanced Two Level Tree WSN can tolerate is f, without change in the total energy consumed.

The Balanced Two Level Tree WSN energy consumption before failure is:

$$Tmax = \left(\frac{N}{\sqrt{N}} - 1\right)Tcomp + \left(\sqrt{N} - 1\right)Tcomp + 2Tnet$$

The Balanced Two Level Tree WSN energy consumption after f failures,

$$Tmax^f = \left(\frac{N}{\sqrt{N} - f} - 1\right)Tcomp + \left(\sqrt{N} - f - 1\right)Tcomp + 2Tnet$$

For energy consumption to be the same even before and after failure, Tmax should be equal to $Tmax^f$.

$$\left(\frac{N}{\sqrt{N}} - 1\right) + \left(\sqrt{N} - 1\right) = \left(\frac{N}{\sqrt{N} - f} - 1\right) + \left(\sqrt{N} - f - 1\right)$$

$$\sqrt{N} = \frac{N}{\sqrt{N}-f} - f$$

which contradicts the assumption. So the IN failures will increase the energy consumption.

## 6.2 Progressive Two Level Tree WSN

IN failure in a ProgressiveTwo Level Tree WSN also follows the reorganization algorithm. Here when one IN fails, the parameters will change from N, m, n1 to N, m-1, n1’ where n1’ is the amount of SNs attached to first IN when one IN fails. Similarly when f IN’s fail, the parameters will change from N, m, n1 to N, m-f, $n1^f$.

Consider a Progressive Two Level Tree WSN n1, n2, n3.... $n_m$ with N total amount of SNs and m of INs.A Progressive Two Level Tree WSNwill have

$$Tmax = (n1 + m - 2)Tcomp + 2\ Tnet$$

When one IN fails in a Progressive Two Level Tree WSN, the new amount of IN will be m-1 and let n1’ be the new n1. So Tmax will be :

$$Tmax = (n1' + m - 3)Tcomp + 2Tnet$$

When ‘f‘ INs fail, the Tmax will be :

$$Tmax = \left(n1^f + m - f - 2\right)Tcomp + 2Tnet \quad (8)$$

where $n1^f$ is the amount of SNs attached to first IN when the amount of INs is m-f.

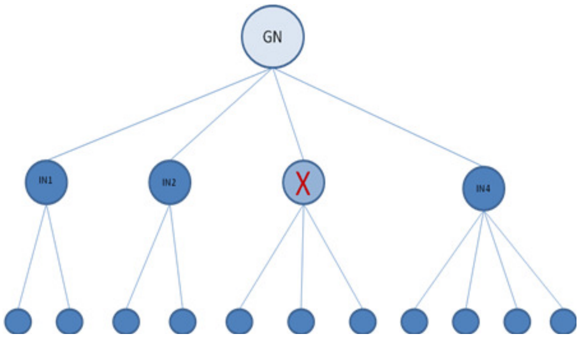



Figure 6: Progressive Two Level Tree WSN before IN Failure

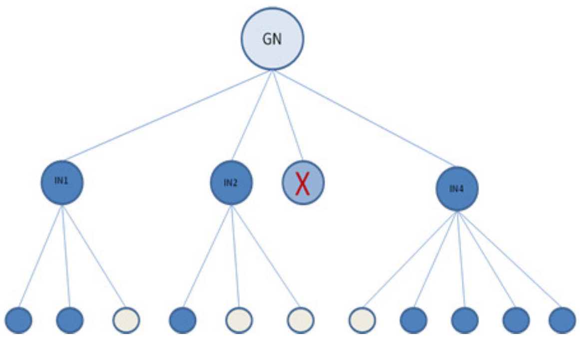



Figure 7: Progressive Two Level Tree WSN after IN failure and reorganization

**Theorem 2**: The energy consumption of a ProgressiveTwo Level Tree WSNincreases as the amount of IN failures increase. For the successful operation of the network we must tolerate IN failures, however the network life time will be compromised.

*Proof*: Let us assume that the amount of IN failures that the ProgressiveTwo Level Tree WSNcan tolerate is f, without change in total energy consumed.

The ProgressiveTwo Level Tree WSNenergy consumption before failure is:

$$Tmax = (n1 + m - 2)Tcomp + 2\ Tnet$$

The ProgressiveTwo Level Tree WSNenergy consumption after f failures, given $n1^f$ is the amount of SNs attached to first IN.

$$Tmax = (n1^f + m - f - 2)Tcomp + 2Tnet$$

For energy consumption to be the same even before and after failure, Tmax should be equal to Tmax'.

$$n1 + m - 2 = n1^f + m - f - 2$$

$$f = n1^f - n1$$

$$\text{i.e } f = m - 2\left(\frac{N-1}{m}\right)$$

where $n1 = \left(\frac{N-1}{m}\right) + \left(\frac{3-m}{2}\right)$ and

$$n1' = \left(\frac{N-1}{m-f}\right) + \left(\frac{3-m+f}{2}\right)$$

But f = m – 2(N-1)/m is negative, which contradicts the assumption. So the failures to IN nodes will increase the energy consumption as the number of computations involved is increased.

# 7. RESULTS AND OBSERVATIONS

## 7.1 Comparison of the Energy Consumption of a Balanced and a Progressive Two Level Tree WSN

The Balanced and the ProgressiveTwo Level Tree WSNswere simulated in MATLAB. In MATLAB, both WSNs were configured with 100, 500, 1000, 1500 of SNs. The network area considered is a square with an area of 1000 x 1000 for 100 and 500 of SNs, and 3000 x 3000 for 1000 and 1500 of SNs. All the SNs are randomly distributed in the network area. They have the same transmission range of 1500m and interference range of 3000m. The algorithm is carried out 20 times, the average value of the results is given as the simulation result. The results were analysed with regard to the amount of SNs.

The results obtained show that Progressive Two Level Tree WSNsare better than Balanced Two Level Tree WSNsas Progressive Two Level Tree WSNs completed their data aggregation first. Therefore ProgressiveTwo Level Tree WSNs are more suitable for timely critical applications as they complete their processing faster than Balanced Two Level Tree WSNs. The aggregation delay is proportional to the amount of computations, which inturn is proportional to the energy consumed. So, Progressive Two Level Tree WSNsconsumes less energy than Balanced Two Level Tree WSNs.

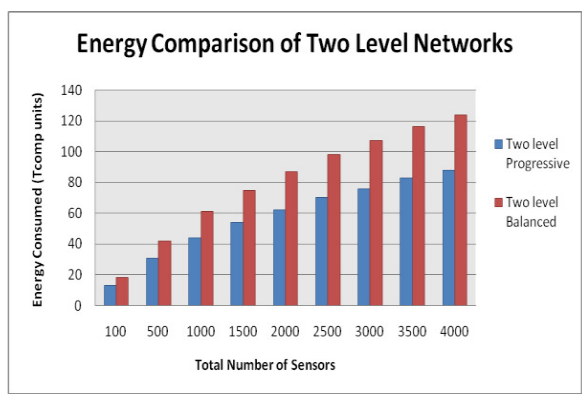



Figure 8: Energy Comparison of Balanced and Progressive Two Level Tree WSNs

Figure 8 shows the Energy Comparison of Balanced and Progressive Two Level Tree WSNs containing up to 4000 SNs.

## 7.2 Sensor Node (SN) Failure

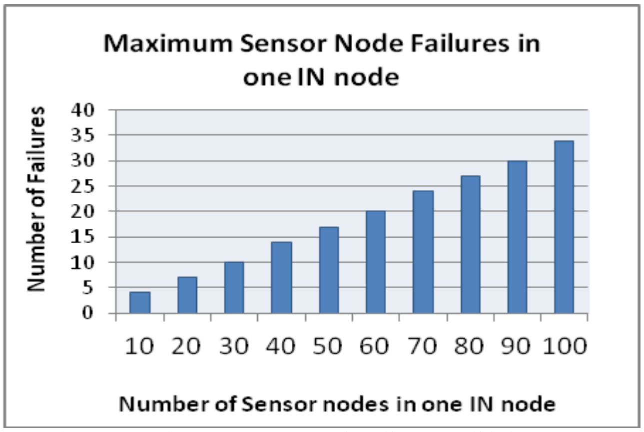



Figure 9: Maximum amount of Node Failures possible in one IN

The amount of failures tolerable to one IN is shown in Figure 9, with different amounts of SNs attached to the same IN. This is subjected to the constraint that between two failed SNs there exists at least two functioning SNs.

The maximum tolerable number of SN failures in a WSN for different N values is shown in Figure 10. The maximum tolerable number of SN failures that each IN in the WSN can tolerate. The total number of SN failures that the WSN can tolerate is equal to the sum of SN failures that each IN can tolerate. The result shows that a Progressive Two Level Tree WSN is capable of tolerating more SN failures than a Balanced Two Level Tree WSN.

## 7.3 Intermediate Node (IN) Failure

The Simulation of IN failures using MATLAB shows that, as the IN failure increases the energy consumption of the whole WSN increases. This enables us to conclude that WSN life time decreases as IN failure increases.

Figure 11 shows that for each N value, the energy consumption of the WSN varies with respect to the amount of INs that the WSN supports. As the amount of INs decreases, the WSN consumes more energy and the total lifetime of the WSN decreases i.e as IN failures increase, energy consumption also

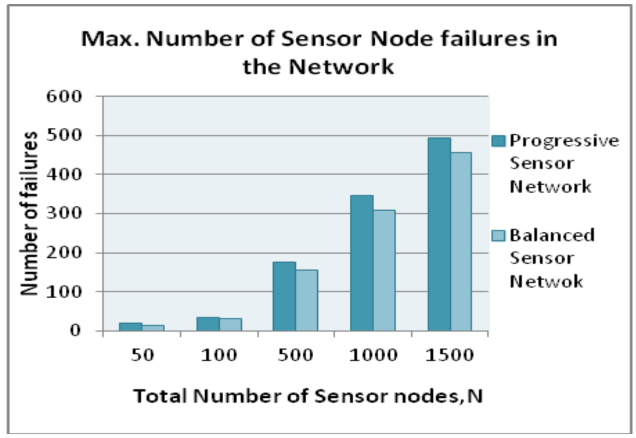



Figure 10: Maximum possible SN Failures in Balanced and Progressive Two Level Tree WSN

increases. The simulation results show that the energy consumption for each IN failure in a Balanced Two Level Tree WSN is more than in a Progressive Two Level Tree WSN.

Assume that the network has 100% lifetime with all IN working with N amount of SNs and m of INs. Let Tmax be the energy consumption with 100% lifetime. When one IN fails, let Tmax' be the new energy consumption. Then the failure of one IN will decrease the energy by $100\left(\frac{Tmax}{Tmax'}-1\right)\%$ and failure of two IN will decrease the energy by $100\left(\frac{Tmax}{Tmax^{2}}-1\right)\%$ and so on. When f INs fails, the energy depletion of the network is by $100\left(\frac{Tmax}{Tmax^{f}}-1\right)\%$. This can be concluded as the energy depletion,due to the amount of IN failures, is dependent on the N, m and f values of the WSN. With respect to its original lifetime, the WSN is assumed to tolerate a ∝% lifetime decrease. And the value of the ∝can be chosen based on the application and its lifetime requirement.

The Figure 12 shows how many IN failures each N-network will tolerate, assuming different values for α. The results show that

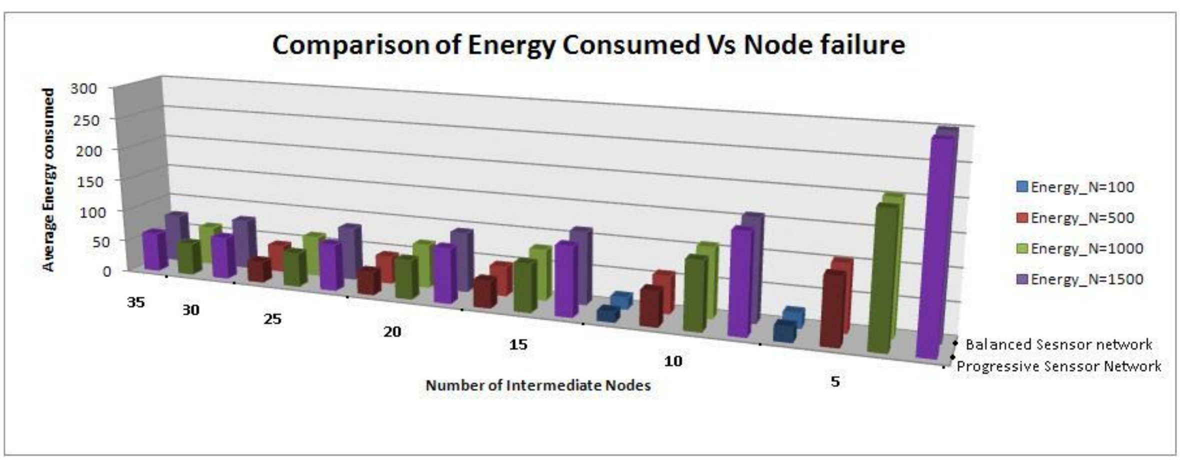



Figure 11: Comparison of Energy Consumed Vs Intermediate Node (IN) Failure in Balanced and Progressive Two Level Tree WSN

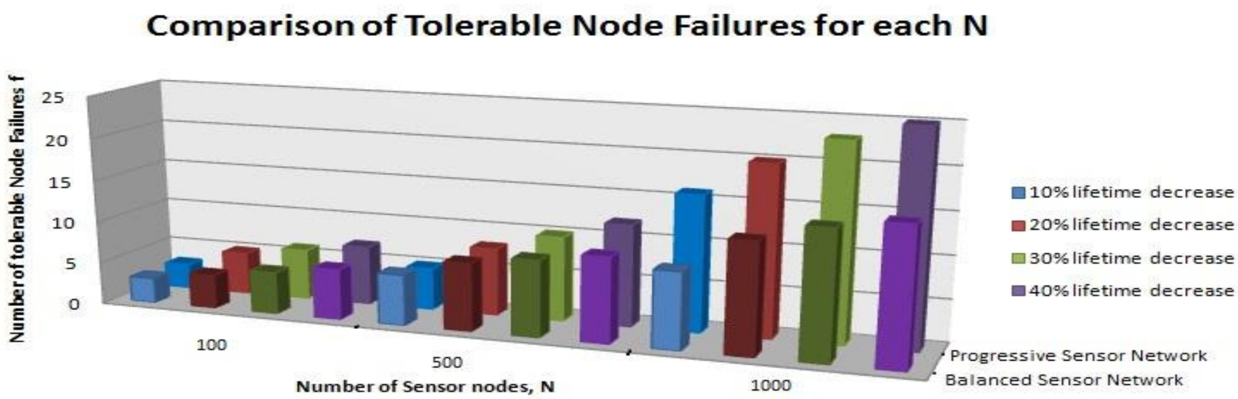



Figure 12: Comparison of Tolerable IN Failures Vs Total number of SNs for Balanced and Progressive Two Level Tree WSN

failures needs to be less. As IN failures increase, network lifetime decreases.

## 8. CONCLUSION

We hypothesised that if we carefully adjust the configuration of a tree WSN, then we can reduce the energy consumed by the whole WSN. The concept backing our hypothesis was that as the number of computations and transmissions is reduced, the energy consumed by the processing and the radio system will also reduce, which in turn reduces the overall energy consumed by the WSN.

We compared simulation results of a Progressive Two Level Tree WSN and a Balanced Two Level Tree WSN. Our simulation results showed that, the progressive configuration required fewer computations (thus having a longer lifetime) than the balanced configuration. Our results prove that as the number of computations in a Progressive Two Level Tree WSN is less, then the energy consumed for processing is also less.

Therefore based on our results, we conclude that a progressive configuration is the optimum choice compared to a balanced configuration when designing a Two Level Tree WSN. For choosing a progressive configuration over a balanced configuration, our results show two advantages which are less computations required to complete each process and more tolerance of node failures. These progressive configuration advantages both lead to more energy efficiency. Our results prove that the configuration, be it balanced or progressive makes a difference when making decisions about designing a tree WSN. Current WSNs are mainly being configured with a balanced tree structure as oppose to a progressive tree structure. However considering our results, we believe that a progressive tree structure will have less delay and save more power than a balanced tree structure.

In the future, the case of Intermediate Nodes (INs) and Sensor Nodes (SNs) failing together in a two level tree WSN will be addressed.

## 9. ACKNOWLWDGEMENT

We would like to express our immense gratitude to our beloved Chancellor Shri. Mata Amritanandamayi Devi for providing motivation and inspiration for doing this research work.

## 10. REFERENCES

[1] M. Yilmaz, “Duty cycle control in wireless sensor networks,” A Thesis Submitted To The Graduate School Of Natural And Applied Sciences Of Middle East Technical University, September 2007.

[2] R. K. Rajgopal Kannan and S. S. Iyengar, “Energy and rate based mac protocol for wireless sensor networks,” SIGMOD Record, vol. 32, pp. 46 – 55, December 2003.

[3] K. Akkaya and M. Younis, “A survey on routing protocols for wireless sensor networks” , Elsevier Ad Hoc Networks 3 (2005) 325–349.

[4] N. H. A. Ababneh, “Energy efficient topology control algorithm for wireless sensor networks,” A thesis submitted at The University of Sydney, vol. 1, pp. 1–177, November 2008.

[5] J. W. Fonda, “Energy efficient wireless sensor network protocols for monitoring and prognostics of large scale systems,” A Dissertation Presented To The Faculty Of The Graduate School Of The University Of Missouri-Rolla, vol. 1, pp. 10 – 211, 2008

[6] Pottie G.J, ‘Wireless sensor networks’, IEEE Information Theory Workshop, 1998

[7] Wee Peng Tay, John N. Tsitsiklis , ‘On the Impact of Node Failures and Unreliable Communications in Dense Sensor Networks’, IEEE Transactions On Signal Processing, Vol. 56, No. 6, June 2008.